
NLSY97 Appendix 5:
Income and Assets Variable Creation

HOUSEHOLD INCOME AND ASSETS

Variables Created: CV_HH_NET_WORTH_Y
CV_INCOME_GROSS_YR
CV_HH_POV_RATIO

Variables Used

Name in Program	Question Name on CD	Name in Program	Question Name on CD
YAS50	YAST-050	YAS4790-YAS4816	YAST-4790-YAST-4816
YAS200, YAS210	YAST-200, YAST-210	YAS4814L, YAS4814U	YAST-4814-000001, ~000002
YAS600	YAST-600	YAS4840-YAS4846	YAST-4840-YAST-4846
YAS900	YAST-900	YAS4844L, YAS4844U	YAST-4844-000001, ~000002
YAS1400, YAS1500	YAST-1400, YAST-1500	YAS4870-YAS4906	YAST-4870-YAST-4906
YAS1610	YAST-1610	YAS4904L, YAS4904U	YAST-4904-000001, ~000002
YAS1860-YAS1866	YAST-1860, YAST-1862, YAST-1866	YAS5040	YAST-5040
YAS2520	YAST-2520	YAS5060, YAS5062	YAST-5060, YAST-5062
YAS2550-YAS2556	YAST-2550, YAST-2552, YAST-2556	A506601-A506620	YAST-5066.01-.20
YAS2760, YAS2766	YAST-2760, YAST-2766	A507401-A507420	YAST-5074.01-.20
YAS3380-YAS3386	YAST-3380, YAST-3382, YAST-3386	A508001-A508020	YAST-5080.01-.20
YAS3384L, YAS3384U	YAST-3384-000001, ~000002	A508201-A508220	YAST-5082.01-.20
YAS3740- YAS3756	YAST-3740-YAST-3756	A508601, A508619	YAST-5086.01, .19
YAS3754L, YAS3754U	YAST-3754-000001, ~000002	YAS5150, YAS5152	YAST-5150, YAST-5152
YAS3780	YAST-3780	A516401-A516420	YAST-5164.01-.20
YAS3790	YAST-3790	A517001-A517020	YAST-5170.01-.20
YAS3810, YAS3812	YAST-3810, YAST-3812	A517201-A517220	YAST-5172.01-.20
YAS3840-YAS3882	YAST-3840-YAST-3882	A517601	YAST-5176.01
YAS3910-YAS3922	YAST-3910-YAST-3922	YAS5210-YAS5226	YAST-5210-YAST-5226
YAS3924L, YAS3924U	YAST-3924-000001, ~000002	YAS5224L, YAS5224U	YAST-5224-000001, ~000002
YAS3950-YAS3966	YAST-3950-YAST-3966	HIRELY01-HIRELY14	HHI_RELY.01-.14
YAS3964L, YAS3964U	YAST-3964-000001, ~000002	YI_300	YINC-300
YAS4010-YAS4036	YAST-4010-YAST-4036	YI_400	YINC-400
YAS4034L, YAS4034U	YAST-4034-000001, ~000002	YI_1400-YI_3100	YINC-1400-YINC-3100
YAS4140-YAS4162	YAST-4140-YAST-4162	YI_3300-YI_5300	YINC-3300-YINC-5300
YAS4270-YAS4296	YAST-4270-YAST-4296	YI_55001-YI_55007	YINC-5500-000000 - ~000006
YAS4294L, YAS4294U	YAST-4294-000001, ~000002	YI_5600-YI_10800	YINC-5600-YINC-10800
YAS4400-YAS4426	YAST-4400-YAST-4426	I1110001-I1110014	YINC-11100.01-.14
YAS4424L, YAS4424U	YAST-4424-000001, ~000002	I1160001-I1160013	YINC-11600.01-.13
YAS4530-YAS4556	YAST-4530-YAST-4556	I1170001-I1170013	YINC-11700.01-.13
YAS4554L, YAS4554U	YAST-4554-000001, ~000002	PUBID	PUBID
YAS4660-YAS4686	YAST-4660-YAST-4686	hhszie	CV_HH_SIZE
YAS4684L, YAS4684U	YAST-4684-000001, ~000002	under18	CV_HH_UNDER_18

This program creates the household net worth and gross household income variables. The household net worth variable is an actual number that results from adding the values of all assets and subtracting liabilities of the household. The gross household income variable includes total annual cash receipts before taxes from all sources. The program then creates a ratio comparing the household's total income to federal poverty guidelines based on the number of household residents and the number of members under age 18.

Researchers should note that, like many income and asset variables in the data set, these three variables are topcoded to protect respondent privacy. More information about topcoding is available in the *NLSY97 User's Guide*.

******* SECTION 1: GROSS HOUSEHOLD INCOME *******

```

flag=0;
/*First, create a variable indicating net receipts from non-farm employment earned by youth (Y), such as wages
(nfarmwgY). */
nfarmwgY=-4;
flag=0;
if (YI_1400=1 and YI_1700 ne -1 and YI_1700 ne -2 and YI_1700 ne -3)

```

Appendix 5: Income and Assets Variable Creation

```

or (YI_1400=-1 and YI_1600=1 and YI_1700 ne -1 and YI_1700 ne -2 and YI_1700 ne -3)
or (YI_1400=-2 and YI_1500=1 and YI_1700 ne -1 and YI_1700 ne -2 and YI_1700 ne -3)
or (YI_1400=-2 and YI_1500=-1 and YI_1600=1 and YI_1700 ne -1 and YI_1700 ne -2 and YI_1700 ne -3)
    then nfarmwgY=YI_1700;

if (YI_1400=1 or (YI_1400=-1 and YI_1600=1) or (YI_1400=-2 and YI_1500=1) or (YI_1400=-2 and YI_1500=-1
and YI_1600=1) or (YI_1400=-3 and YI_1600=1) or (YI_1400=-3 and YI_1500=1) or (YI_1400=-2 and YI_1500=-3
and YI_1600=1)) and (YI_1700 eq -1 or YI_1700 eq -2 or YI_1700 eq -3) then do;
    if YI_1800=1 then do; nfarmwgY=2500; flag=1; end;
    if YI_1800=2 then do; nfarmwgY=7500; flag=1; end;
    if YI_1800=3 then do; nfarmwgY=17500; flag=1; end;
    if YI_1800=4 then do; nfarmwgY=37500; flag=1; end;
    if YI_1800=5 then do; nfarmwgY=75000; flag=1; end;
    if YI_1800=6 then do; nfarmwgY=175000; flag=1; end;
    if YI_1800=7 then do; nfarmwgY=250001; flag=1; end;
end;
if YI_1400=0 or YI_1500=0 or YI_1600=0 then nfarmwgY=0;
if YI_1600=-1 or YI_1800=-1 then nfarmwgY=-1;
if YI_1500=-2 or YI_1800=-2 then nfarmwgY=-2;
if YI_1600=-3 or YI_1500=-3 or YI_1800=-3 then nfarmwgY=-3;

/* For all the questions below, the youth must be INDEPENDENT (YI_1900=1) */

/*Second, create a variable indicating net receipts from farm self-employment earned by the Y (farmwgY)*/
farmwgY=-4;
if YI_2000=1 and YI_2100 ne -1 and YI_2100 ne -2 and YI_2100 ne -3 then farmwgY=YI_2100;
if YI_2000=1 and (YI_2100 eq -1 or YI_2100 eq -2 or YI_2100 eq -3) then do;
    if YI_2200=1 then do; farmwgY=-2; flag=1; end;
    if YI_2200=2 then do; farmwgY=2500; flag=1; end;
    if YI_2200=3 then do; farmwgY=7500; flag=1; end;
    if YI_2200=4 then do; farmwgY=17500; flag=1; end;
    if YI_2200=5 then do; farmwgY=37500; flag=1; end;
    if YI_2200=6 then do; farmwgY=75000; flag=1; end;
    if YI_2200=7 then do; farmwgY=175000; flag=1; end;
    if YI_2200=8 then do; farmwgY=250001; flag=1; end;
end;
if YI_2000=0 then farmwgY=0;
if YI_2000=-1 or YI_2200=-1 then farmwgY=-1;
if YI_2000=-2 or YI_2200=-2 then farmwgY=-2;
if YI_2000=-3 or YI_2200=-3 then farmwgY=-3;

/*Third, create the above variables for the event the Y has a spouse/partner: nfarmwgP and farmwgP respectively */

nfarmwgP=-4;
if (YI_2300=1 and YI_2400=1 and YI_2600 ne -1 and YI_2600 ne -2 and YI_2600 ne -3) or
(YI_2300=1 and YI_2400=-1 and YI_2500=1 and YI_2600 ne -1 and YI_2600 ne -2 and YI_2600 ne -3)
    then nfarmwgP=YI_2600;
if (YI_2300=1 and YI_2400=1 and (YI_2600 eq -1 or YI_2600 eq -2 or YI_2600 eq -3)) or
(YI_2300=1 and YI_2400=-1 and YI_2500=1 and (YI_2600 eq -1 or YI_2600 eq -2 or YI_2600 eq -3)) then do;
    if YI_2700=1 then do; nfarmwgP=2500; flag=1; end;
    if YI_2700=2 then do; nfarmwgP=7500; flag=1; end;
    if YI_2700=3 then do; nfarmwgP=17500; flag=1; end;
    if YI_2700=4 then do; nfarmwgP=37500; flag=1; end;
    if YI_2700=5 then do; nfarmwgP=75000; flag=1; end;
    if YI_2700=6 then do; nfarmwgP=175000; flag=1; end;
    if YI_2700=7 then do; nfarmwgP=250001; flag=1; end;
end;

```

```

if YI_2300=0 or YI_2400=0 or YI_2500=0 then nfarmwgP=0;
if YI_2300=-1 or YI_2500=-1 or YI_2700=-1 then nfarmwgP=-1;
if YI_2300=-2 or YI_2400=-2 or YI_2500=-2 or YI_2700=-2 then nfarmwgP=-2;
if YI_2300=-3 or YI_2400=-3 or YI_2500=-3 or YI_2700=-3 then nfarmwgP=-3;

/* For any farm-related income from the spouse or partner */
farmwgP=-4;
if YI_2300=1 and YI_2900=1 and YI_3000 ne -1 and YI_3000 ne -2 and YI_3000 ne -3 then farmwgP=YI_3000;
if YI_2300=1 and YI_2900=1 and (YI_3000 eq -1 or YI_3000 eq -2 or YI_3000 eq -3) then do;
    if YI_3100=1 then do; farmwgP=-2; flag=1; end;
    if YI_3100=2 then do; farmwgP=2500; flag=1; end;
    if YI_3100=3 then do; farmwgP=7500; flag=1; end;
    if YI_3100=4 then do; farmwgP=17500; flag=1; end;
    if YI_3100=5 then do; farmwgP=37500; flag=1; end;
    if YI_3100=6 then do; farmwgP=75000; flag=1; end;
    if YI_3100=7 then do; farmwgP=175000; flag=1; end;
    if YI_3100=8 then do; farmwgP=250001; flag=1; end;
end;
if YI_2300=0 or YI_2900=0 then farmwgP=0;
if YI_2300=-1 or YI_2900=-1 or YI_3100=-1 then farmwgP=-1;
if YI_2300=-2 or YI_2900=-2 or YI_3100=-2 then farmwgP=-2;
if YI_2300=-3 or YI_2900=-3 or YI_3100=-3 then farmwgP=-3;

/* create a variable indicating whether they collected any child support (childsuY) */
childsuY=-4;
if YI_3300=1 and YI_3900=1 and YI_4000=1 and YI_4100 ne -1 and YI_4100 ne -2 and YI_4100 ne -3
    then childsuY=YI_4100;
if YI_3300=1 and YI_3900=1 and YI_4000=1 and (YI_4100 eq -1 or YI_4100 eq -2 or YI_4100 eq -3) then do;
    if YI_4200=1 then do; childsuY=500; flag=1; end;
    if YI_4200=2 then do; childsuY=1750; flag=1; end;
    if YI_4200=3 then do; childsuY=3750; flag=1; end;
    if YI_4200=4 then do; childsuY=7500; flag=1; end;
    if YI_4200=5 then do; childsuY=17500; flag=1; end;
    if YI_4200=6 then do; childsuY=37500; flag=1; end;
    if YI_4200=7 then do; childsuY=50001; flag=1; end;
end;
if YI_3300=0 or YI_3900=0 or YI_4000=0 then childsuY=0;
if YI_3300=-1 or YI_3900=-1 or YI_4000=-1 or YI_4200=-1 then childsuY=-1;
if YI_3300=-2 or YI_3900=-2 or YI_4000=-2 or YI_4200=-2 then childsuY=-2;
if YI_3300=-1 or YI_3900=-1 or YI_4000=-3 or YI_4200=-3 then childsuY=-3;

/* create a variable indicating the amount of interest received by the youth (Y) and his/her partner/spouse */
interesY=-4;
if YI_4300=1 and YI_4400 ne -1 and YI_4400 ne -2 and YI_4400 ne -3 then interesY=YI_4400;
if YI_4300=1 and (YI_4400 eq -1 or YI_4400 eq -2 or YI_4400 eq -3) then do;
    if YI_4500=1 then do; interesY=250; flag=1; end;
    if YI_4500=2 then do; interesY=750; flag=1; end;
    if YI_4500=3 then do; interesY=1750; flag=1; end;
    if YI_4500=4 then do; interesY=3750; flag=1; end;
    if YI_4500=5 then do; interesY=6250; flag=1; end;
    if YI_4500=6 then do; interesY=8750; flag=1; end;
    if YI_4500=7 then do; interesY=10001; flag=1; end;
end;
if YI_4300=0 then interesY=0;
if YI_4300=-1 or YI_4500=-1 then interesY=-1;

```

Appendix 5: Income and Assets Variable Creation

```
if YI_4300=-2 or YI_4500=-2 then interesY=-2;
if YI_4300=-3 or YI_4400=-3 or YI_4500=-3 then interesY=-3;

/* create a variable indicating whether they collected any dividends from stocks and mutual funds */
dividend=-4;
if YI_4600=1 and YI_4700 ne -1 and YI_4700 ne -2 and YI_4700 ne -3 then dividend=YI_4700;
if YI_4600=1 and (YI_4700 eq -1 or YI_4700 eq -2 or YI_4700 eq -3) then do;
    if YI_4800=1 then do; dividend=250; flag=1; end;
    if YI_4800=2 then do; dividend=750; flag=1; end;
    if YI_4800=3 then do; dividend=1750; flag=1; end;
    if YI_4800=4 then do; dividend=3750; flag=1; end;
    if YI_4800=5 then do; dividend=6250; flag=1; end;
    if YI_4800=6 then do; dividend=8750; flag=1; end;
    if YI_4800=7 then do; dividend=10001; flag=1; end;
end;
if YI_4600=0 then dividend=0;
if YI_4600=-1 or YI_4800=-1 then dividend=-1;
if YI_4600=-2 or YI_4800=-2 then dividend=-2;
if YI_4600=-3 or YI_4700=-3 or YI_4800=-3 then dividend=-3;

/* create a variable indicating any rental income */
rentalIY=-4;
if YI_4900=1 and YI_5000 ne -1 and YI_5000 ne -2 and YI_5000 ne -3 then rentalIY=YI_5000;
if YI_4900=1 and (YI_5000 eq -1 or YI_5000 eq -2 or YI_5000 eq -3) then do;
    if YI_5100=1 then do; rentalIY=500; flag=1; end;
    if YI_5100=2 then do; rentalIY=1750; flag=1; end;
    if YI_5100=3 then do; rentalIY=3750; flag=1; end;
    if YI_5100=4 then do; rentalIY=7500; flag=1; end;
    if YI_5100=5 then do; rentalIY=17500; flag=1; end;
    if YI_5100=6 then do; rentalIY=37500; flag=1; end;
    if YI_5100=7 then do; rentalIY=50001; flag=1; end;
end;
if YI_4900=0 then rentalIY=0;
if YI_4900=-1 or YI_5100=-1 then rentalIY=-1;
if YI_4900=-2 or YI_5100=-2 then rentalIY=-2;
if YI_4900=-3 or YI_5000=-3 or YI_5100=-3 then rentalIY=-3;

/* create a variable indicating whether they received any property or money from estates, trusts, annuities or
inheritances */

estatesY=-4;
if YI_5200=1 and YI_5300 ne -1 and YI_5300 ne -2 and YI_5300 ne -3 then estatesY=YI_5300;
if YI_5200=1 and (YI_5300 eq -1 or YI_5300 eq -2 or YI_5300 eq -3) then do;
    if YI_5400=1 then do; estatesY=2500; flag=1; end;
    if YI_5400=2 then do; estatesY=7500; flag=1; end;
    if YI_5400=3 then do; estatesY=17500; flag=1; end;
    if YI_5400=4 then do; estatesY=37500; flag=1; end;
    if YI_5400=5 then do; estatesY=75000; flag=1; end;
    if YI_5400=6 then do; estatesY=175000; flag=1; end;
    if YI_5400=7 then do; estatesY=250001; flag=1; end;
end;
if YI_5200=0 then estatesY=0;
if YI_5200=-1 or YI_5400=-1 then estatesY=-1;
if YI_5200=-2 or YI_5400=-2 then estatesY=-2;
if YI_5200=-3 or YI_5300=-3 or YI_5400=-3 then estatesY=-3;
```

```

/* create a variable indicating any allowances parents might have given the youth if he/she still lives with them (NOT INCLUDED IN THE YOUTH INCOME) */
allowpar=-4;
if YI_5600=1 then do;
if YI_5700=1 and YI_5800 ne -1 and YI_5800 ne -2 and YI_5800 ne -3 then allowpar=YI_5800;
if YI_5700=1 and (YI_5800 eq -1 or YI_5800 eq -2 or YI_5800 eq -3) then do;
    if YI_5900=1 then do; allowpar=250; flag=1; end;
    if YI_5900=2 then do; allowpar=750; flag=1; end;
    if YI_5900=3 then do; allowpar=1750; flag=1; end;
    if YI_5900=4 then do; allowpar=3750; flag=1; end;
    if YI_5900=5 then do; allowpar=6250; flag=1; end;
    if YI_5900=6 then do; allowpar=8750; flag=1; end;
    if YI_5900=7 then do; allowpar=10001; flag=1; end;
end;
if YI_5700=0 then allowpar=0;
if YI_5700=-1 or YI_5900=-1 then allowpar=-1;
if YI_5700=-2 or YI_5900=-2 then allowpar=-2;
if YI_5700=-3 or YI_5800=-3 or YI_5900=-3 then allowpar=-3;
end;

/* If living with your mother or female guardian, did you receive any money from her? (NOT INCLUDED IN THE YOUTH INCOME DEFINITION) */
allowmot=-4;
if (HIRELY01 ne 3 and HIRELY02 ne 3 and HIRELY03 ne 3 and HIRELY04 ne 3 and HIRELY05 ne 3 and HIRELY06 ne 3 and HIRELY07 ne 3 and HIRELY08 ne 3 and HIRELY09 ne 3 and HIRELY10 ne 3 and HIRELY11 ne 3 and HIRELY12 ne 3 and HIRELY13 ne 3 and HIRELY14 ne 3) then do;
    if YI_6500=1 and YI_6600 ne -1 and YI_6600 ne -2 and YI_6600 ne -3 then allowmot=YI_6600;
    if YI_6500=1 and (YI_6600 eq -1 or YI_6600 eq -2 or YI_6600 eq -3) then do;
        if YI_6700=1 then do; allowmot=250; flag=1; end;
        if YI_6700=2 then do; allowmot=750; flag=1; end;
        if YI_6700=3 then do; allowmot=1750; flag=1; end;
        if YI_6700=4 then do; allowmot=3750; flag=1; end;
        if YI_6700=5 then do; allowmot=6250; flag=1; end;
        if YI_6700=6 then do; allowmot=8750; flag=1; end;
        if YI_6700=7 then do; allowmot=10001; flag=1; end;
    end;
if YI_6500=0 then allowmot=0;
if YI_6500=-1 or YI_6700=-1 then allowmot=-1;
if YI_6500=-2 or YI_6700=-2 then allowmot=-2;
if YI_6500=-3 or YI_6600=-3 or YI_6700=-3 then allowmot=-3;
end;

/* If the youth lived with the father or male guardian and received any allowances from him (NOT INCLUDED IN THE YOUTH INCOME DEFINITION) */
allowfat=-4;
if (HIRELY01 ne 4 and HIRELY02 ne 4 and HIRELY03 ne 4 and HIRELY04 ne 4 and HIRELY05 ne 4 and HIRELY06 ne 4 and HIRELY07 ne 4 and HIRELY08 ne 4 and HIRELY09 ne 4 and HIRELY10 ne 4 and HIRELY11 ne 4 and HIRELY12 ne 4 and HIRELY13 ne 4 and HIRELY14 ne 4) then do;
    if YI_7100=1 and YI_7200 ne -1 and YI_7200 ne -2 and YI_7200 ne -3 then allowfat=YI_7200;
    if YI_7100=1 and (YI_7200 eq -1 or YI_7200 eq -2 or YI_7200 eq -3) then do;
        if YI_7300=1 then do; allowfat=250; flag=1; end;
        if YI_7300=2 then do; allowfat=750; flag=1; end;
        if YI_7300=3 then do; allowfat=1750; flag=1; end;
        if YI_7300=4 then do; allowfat=3750; flag=1; end;
        if YI_7300=5 then do; allowfat=6250; flag=1; end;
        if YI_7300=6 then do; allowfat=8750; flag=1; end;
        if YI_7300=7 then do; allowfat=10001; flag=1; end;
    end;

```

```

    end;
if YI_7100=0 then allowfat=0;
if YI_7100=-1 or YI_7300=-1 then allowfat=-1;
if YI_7100=-2 or YI_7300=-2 then allowfat=-2;
if YI_7100=-3 or YI_7200=-3 or YI_7300=-3 then allowfat=-3;
end;

/* Income received by the youth from other sources: SS payments, pension and retirement income, alimony,
payments from insurance policies, etc... */
pensionY=-4;
if YI_7600=1 and YI_7700 ne -1 and YI_7700 ne -2 and YI_7700 ne -3 then pensionY=YI_7700;
if YI_7600=1 and (YI_7700 eq -1 or YI_7700 eq -2 or YI_7700 eq -3) then do;
    if YI_7800=1 then do; pensionY=500; flag=1; end;
    if YI_7800=2 then do; pensionY=1750; flag=1; end;
    if YI_7800=3 then do; pensionY=3750; flag=1; end;
    if YI_7800=4 then do; pensionY=7500; flag=1; end;
    if YI_7800=5 then do; pensionY=17500; flag=1; end;
    if YI_7800=6 then do; pensionY=37500; flag=1; end;
    if YI_7800=7 then do; pensionY=50001; flag=1; end;
end;
if YI_7600=0 then pensionY=0;
if YI_7600=-1 or YI_7800=-1 then pensionY=-1;
if YI_7600=-2 or YI_7800=-2 then pensionY=-2;
if YI_7600=-3 or YI_7700=-3 or YI_7800=-3 then pensionY=-3;

/* For everyone, DEPENDENT OR INDEPENDENT, allowances received by the youth from his/her family
(NOT INCLUDED IN THE INCOME DEFINITION OF THE YOUTH) */
yfaallow=-4;
if YI_8100=1 and YI_8200 ne -1 and YI_8200 ne -2 and YI_8200 ne -3 and YI_8300=1 then yfaallow=YI_8200*52;
if YI_8100=1 and YI_8200 ne -1 and YI_8200 ne -2 and YI_8200 ne -3 and YI_8300=2 then yfaallow=YI_8200*12;
if YI_8100=1 and YI_8200 ne -1 and YI_8200 ne -2 and YI_8200 ne -3 and YI_8300=3 then do;
    yfaallow=-3; flag=1; end;
if YI_8100=0 then yfaallow=0;
if YI_8100=-1 or YI_8200=-1 or YI_8300=-1 then yfaallow=-1;
if YI_8100=-2 or YI_8200=-2 or YI_8300=-2 then yfaallow=-2;
if YI_8100=-3 or YI_8200=-3 or YI_8300=-3 then yfaallow=-3;

/* Now a few more questions regarding income if the individual is INDEPENDENT (YI_8500=1) or 14 YEARS OF
AGE OR OLDER (YI_8400=1)*

/* If the youth lived with the father, the father's income */
faincome=-4;
/* identifying the father is in the household */
if (HIRELY01=4 or HIRELY02=4 or HIRELY03=4 or HIRELY04=4 or HIRELY05=4 or HIRELY06=4 or
HIRELY07=4 or HIRELY08=4 or HIRELY09=4 or HIRELY10=4 or HIRELY11=4 or HIRELY12=4 or
HIRELY13=4 or HIRELY14=4) then do;
    if YI_8700=1 and YI_8800 ne -1 and YI_8800 ne -2 and YI_8800 ne -3 then faincome=YI_8800;
    if YI_8700=1 and (YI_8800 eq -1 or YI_8800 eq -2 or YI_8800 eq -3) then do;
        if YI_8900=1 then do; faincome=2500; flag=1; end;
        if YI_8900=2 then do; faincome=7500; flag=1; end;
        if YI_8900=3 then do; faincome=17500; flag=1; end;
        if YI_8900=4 then do; faincome=37500; flag=1; end;
        if YI_8900=5 then do; faincome=75000; flag=1; end;
        if YI_8900=6 then do; faincome=175000; flag=1; end;
        if YI_8900=7 then do; faincome=250001; flag=1; end;
    end;
end;

```

```

if YI_8700=0 then faincome=0;
if YI_8700=-1 or YI_8900=-1 then faincome=-1;
if YI_8700=-2 or YI_8900=-2 then faincome=-2;
if YI_8700=-3 or YI_8800=-3 or YI_8900=-3 then faincome=-3;

/* If the youth lived with the mother, the mother's income */
maincome=-4;
/* identify the mother is in the household */
if (HIRELY01=3 or HIRELY02=3 or HIRELY03=3 or HIRELY04=3 or HIRELY05=3 or HIRELY06=3 or
HIRELY07=3 or HIRELY08=3 or HIRELY09=3 or HIRELY10=3 or HIRELY11=3 or HIRELY12=3 or
HIRELY13=3 or HIRELY14=3) then do;
    if YI_9200=1 and YI_9300 ne -1 and YI_9300 ne -2 and YI_9300 ne -3 then maincome=YI_9300;
    if YI_9200=1 and (YI_9300 eq -1 or YI_9300 eq -2 or YI_9300 eq -3) then do;
        if YI_9400=1 then do; maincome=2500; flag=1; end;
        if YI_9400=2 then do; maincome=7500; flag=1; end;
        if YI_9400=3 then do; maincome=17500; flag=1; end;
        if YI_9400=4 then do; maincome=37500; flag=1; end;
        if YI_9400=5 then do; maincome=75000; flag=1; end;
        if YI_9400=6 then do; maincome=175000; flag=1; end;
        if YI_9400=7 then do; maincome=250001; flag=1; end;
    end;
end;
if YI_9200=0 then maincome=0;
if YI_9200=-1 or YI_9400=-1 then maincome=-1;
if YI_9200=-2 or YI_9400=-2 then maincome=-2;
if YI_9200=-3 or YI_9400=-3 then maincome=-3;

/* If the youth lives with the male guardian */
mgincome=-4;
if YI_9600=1 and YI_9700=1 and YI_9800 ne -1 and YI_9800 ne -2 and YI_9800 ne -3 then mgincome=YI_9800;
if YI_9600=1 and YI_9700=1 and (YI_9800 eq -1 or YI_9800 eq -2 or YI_9800 eq -3) then do;
    if YI_9900=1 then do; mgincome=2500; flag=1; end;
    if YI_9900=2 then do; mgincome=7500; flag=1; end;
    if YI_9900=3 then do; mgincome=17500; flag=1; end;
    if YI_9900=4 then do; mgincome=37500; flag=1; end;
    if YI_9900=5 then do; mgincome=75000; flag=1; end;
    if YI_9900=6 then do; mgincome=175000; flag=1; end;
    if YI_9900=7 then do; mgincome=250001; flag=1; end;
end;
if YI_9700=0 then mgincome=0;
if YI_9600=-1 or YI_9700=-1 or YI_9900=-1 then mgincome=-1;
if YI_9600=-2 or YI_9700=-2 or YI_9900=-2 then mgincome=-2;
if YI_9600=-3 or YI_9700=-3 or YI_9800=-3 or YI_9900=-3 then mgincome=-3;

/* If the youth lives with female guardian */
fgincome=-4;
if YI_10100=1 and YI_10200=1 and YI_10300 ne -1 and YI_10300 ne -2 and YI_10300 ne -3
    then fgincome=YI_10300;
if YI_10100=1 and YI_10200=1 and (YI_10300 eq -1 or YI_10300 eq -2 or YI_10300 eq -3) then do;
    if YI_10400=1 then do; fgincome=2500; flag=1; end;
    if YI_10400=2 then do; fgincome=7500; flag=1; end;
    if YI_10400=3 then do; fgincome=17500; flag=1; end;
    if YI_10400=4 then do; fgincome=37500; flag=1; end;
    if YI_10400=5 then do; fgincome=75000; flag=1; end;
    if YI_10400=6 then do; fgincome=175000; flag=1; end;
    if YI_10400=7 then do; fgincome=250001; flag=1; end;
end;

```

```

if YI_10200=0 then fgincome=0;
if YI_10100=-1 or YI_10200=-1 or YI_10400=-1 then fgincome=-1;
if YI_10100=-2 or YI_10200=-2 or YI_10400=-2 then fgincome=-2;
if YI_10100=-3 or YI_10200=-3 or YI_10300=-3 or YI_10400=-3 then fgincome=-3;

/* Check for income from any household member 14 years of age or older and who hasn't been asked before */

array otfamI otfamI01-otfamI14;
array I11100 I1110001-I1110014;
array I11600 I1160001-I1160014;
array I11700 I1170001-I1170014;

do I=1 to 14;
  otfamI(I)=-4;
  if YI_10800=1 then do;
    if I11100(I)=0 and I11600(I) ne -1 and I11600(I) ne -2 and I11600(I) ne -3 and I11600(I) ne -4
      then otfamI(I)=I11600(I);
    if I11100(I)=0 and (I11600(I) eq -1 or I11600(I) eq -2 or I11600(I) eq -3) then do;
      if I11700(I)=1 then do; otfamI(I)=2500; flag=1; end;
      if I11700(I)=2 then do; otfamI(I)=7500; flag=1; end;
      if I11700(I)=3 then do; otfamI(I)=17500; flag=1; end;
      if I11700(I)=4 then do; otfamI(I)=37500; flag=1; end;
      if I11700(I)=5 then do; otfamI(I)=75000; flag=1; end;
      if I11700(I)=6 then do; otfamI(I)=175000; flag=1; end;
      if I11700(I)=7 then do; otfamI(I)=250001; flag=1; end;
    end;
    if I11100(I)=-1 or I11700(I)=-1 then otfamI(I)=-1;
    if I11100(I)=-2 or I11700(I)=-2 then otfamI(I)=-2;
    if I11100(I)=-3 or I11600(I)=-3 or I11700(I)=-3 then otfamI(I)=-3;
  end;
end;

/* We now create gross hh income according to the youth*/

groshhIY=0;
if YI_1900=1 then do;
  if nfarmwgY not in (-1,-2,-3,-4) then groshhIY=groshhIY+nfarmwgY;
  if farmwgY not in (-1,-2,-3,-4) then groshhIY=groshhIY+farmwgY;
  if nfarmwgP not in (-1,-2,-3,-4) then groshhIY=groshhIY+nfarmwgP;
  if farmwgP not in (-1,-2,-3,-4) then groshhIY=groshhIY+farmwgP;
  if childsuY not in (-1,-2,-3,-4) then groshhIY=groshhIY+childsuY;
  if interesY not in (-1,-2,-3,-4) then groshhIY=groshhIY+interesY;
  if dividend not in (-1,-2,-3,-4) then groshhIY=groshhIY+dividend;
  if rentalIY not in (-1,-2,-3,-4) then groshhIY=groshhIY+rentalIY;
  if estatesY not in (-1,-2,-3,-4) then groshhIY=groshhIY+estatesY;
  if pensionY not in (-1,-2,-3,-4) then groshhIY=groshhIY+pensionY;
  if faincome not in (-1,-2,-3,-4) then groshhIY=groshhIY+faincome;
  if maincome not in (-1,-2,-3,-4) then groshhIY=groshhIY+maincome;
  if mgincome not in (-1,-2,-3,-4) then groshhIY=groshhIY+mgincome;
  if fgincome not in (-1,-2,-3,-4) then groshhIY=groshhIY+fgincome;
  if afdcy not in (-1,-2,-3,-4) then groshhIY=groshhIY+afdcy;
  if ssiy not in (-1,-2,-3,-4) then groshhIY=groshhIY+ssiy;
  if othe not in (-1,-2,-3,-4) then groshhIY=groshhIY+othe;
  if prgamt not in (-1,-2,-3,-4) then groshhIY=groshhIY+prgamt;
  do I=1 to 14;
    if otfamI[I] not in (-1,-2,-3,-4) then groshhIY=groshhIY+otfamI[I];
  end;

```

```

end;

if YI_1900=-1 or YI_8500=-1 or (nfarmwgY=-1 or farmwgY=-1 or nfarmwgP=-1 or farmwgP=-1 or childsuY=-1 or interesY=-1 or dividend=-1 or rentalIY=-1 or estatesY=-1 or pensionY=-1 or faincome=-1 or maincome=-1 or mgincome=-1 or fgincome=-1 or of famI01=-1 or of famI02=-1 or of famI03=-1 or of famI04=-1 or of famI04=-1 or of famI05=-1 or of famI06=-1 or of famI07=-1 or of famI08=-1 or of famI09=-1 or of famI10=-1 or of famI11=-1 or of famI12=-1 or of famI13=-1 or of famI14=-1 or afdcy=-1 or ssiy=-1 or othe=-1 or prgamt=-1) then
    groshhIY=-1;
if YI_1900=-2 or YI_8500=-2 or (nfarmwgY=-2 or farmwgY=-2 or nfarmwgP=-2 or farmwgP=-2 or childsuY=-2 or interesY=-2 or dividend=-2 or rentalIY=-2 or pensionY=-2 or estatesY=-2 or faincome=-2 or maincome=-2 or mgincome=-2 or fgincome=-2 or of famI01=-2 or of famI02=-2 or of famI03=-2 or of famI04=-2 or of famI04=-2 or of famI05=-2 or of famI06=-2 or of famI07=-2 or of famI08=-2 or of famI09=-2 or of famI10=-2 or of famI11=-2 or of famI12=-2 or of famI13=-2 or of famI14=-2 or afdcy=-2 or ssiy=-2 or othe=-2 or prgamt=-2 ) then
    groshhIY=-2;
if YI_1900=-3 or YI_8500=-3 or (nfarmwgY=-3 or farmwgY=-3 or nfarmwgP=-3 or farmwgP=-3 or childsuY=-3 or interesY=-3 or dividend=-3 or rentalIY=-3 or pensionY=-3 or estatesY=-3 or faincome=-3 or maincome=-3 or mgincome=-3 or fgincome=-3 or of famI01=-3 or of famI02=-3 or of famI03=-3 or of famI04=-3 or of famI04=-3 or of famI05=-3 or of famI06=-3 or of famI07=-3 or of famI08=-3 or of famI09=-3 or of famI10=-3 or of famI11=-3 or of famI12=-3 or of famI13=-3 or of famI14=-3 or afdcy=-3 or ssiy=-3 or othe=-3 or prgamt=-3 ) then
    groshhIY=-3;
if YI_1900=0 or YI_1900=-4 then groshhIY=-4;
if YAS50=-5 then groshhIY=-5; /*give the people who were not interviewed in round 2 value -5.*/

```

***** SECTION 2: HOUSEHOLD NET WORTH *****

if YAS50=1 then do;

/* If the youth OWNS some land, ITS PRESENT VALUE. */

```

pvranch=0;
if YAS1400=1 then do;
    if YAS1610=1 or YAS1610=2 then do;
        if YAS1860=1 then pvranch=YAS1862;
        if YAS1860=2 then pvranch=YAS1864L+(YAS1864U-YAS1864L)/2;
        if ((YAS1860 ne 1 and YAS1860 ne 2) or (YAS1862=-1 or YAS1862=-2)) then do;
            if YAS1866=1 then do; pvranch=12500; flag=1; end;
            if YAS1866=2 then do; pvranch=37500; flag=1; end;
            if YAS1866=3 then do; pvranch=75000; flag=1; end;
            if YAS1866=4 then do; pvranch=175000; flag=1; end;
            if YAS1866=5 then do; pvranch=375000; flag=1; end;
            if YAS1866=6 then do; pvranch=750000; flag=1; end;
            if YAS1866=7 then do; pvranch=1000001; flag=1; end;
        end;
    end;
if (YAS1610=3 or YAS1610=4) and YAS2120=100 then do; /* If the youth reported owning only part of it */
    if YAS2140=1 then pvranch=YAS2142;
    if YAS2140=2 then pvranch=(YAS2144L+(YAS2144U-YAS2144L)/2);
    if ((YAS2140 NE 1 AND YAS2140 NE 2) OR (YAS2142=-1 OR YAS2142=-2)) then do;
        if YAS2146=1 then do; pvranch=12500; flag=1; end;
        if YAS2146=2 then do; pvranch=37500; flag=1; end;
        if YAS2146=3 then do; pvranch=75000; flag=1; end;
        if YAS2146=4 then do; pvranch=175000; flag=1; end;
        if YAS2146=5 then do; pvranch=375000; flag=1; end;
        if YAS2146=6 then do; pvranch=750000; flag=1; end;
        if YAS2146=7 then do; pvranch=1000001; flag=1; end;

```

```

    end;
end;
if (YAS1610=3 or YAS1610=4) and YAS2120 ne 100 then do;
    if YAS2170=1 then pvranch=YAS2172;
    if YAS2170=2 then pvranch=(YAS2174L+(YAS2174U-YAS2174L)/2);
    if ((YAS2170 ne 1 and YAS2170 ne 2) or (YAS2172=-1 or YAS2172=-2)) then do;
        if YAS2176=1 then do; pvranch=12500; flag=1; end;
        if YAS2176=2 then do; pvranch=37500; flag=1; end;
        if YAS2176=3 then do; pvranch=75000; flag=1; end;
        if YAS2176=4 then do; pvranch=175000; flag=1; end;
        if YAS2176=5 then do; pvranch=375000; flag=1; end;
        if YAS2176=6 then do; pvranch=750000; flag=1; end;
        if YAS2176=7 then do; pvranch=1000001; flag=1; end;
    end;
end;
end;

if YAS1400=-1 or YAS1610=-1 or YAS1864L=-1 or YAS1864U=-1 or YAS1866=-1 or YAS2144L=-1 or
    YAS2144U=-1 or YAS2146=-1 or YAS2174L=-1 or YAS2174U=-1 or YAS2176=-1 then pvranch=-1;
if YAS1400=-2 or YAS1610=-2 or YAS1864L=-2 or YAS1864U=-2 or YAS1866=-2 or YAS2144L=-2 or
    YAS2144U=-2 or YAS2146=-2 or YAS2174L=-2 or YAS2174U=-2 or YAS2176=-2 then pvranch=-2;
if YAS1400=-3 or YAS1610=-3 or YAS1864L=-3 or YAS1864U=-3 or YAS1866=-3 or YAS2144L=-3 or
    YAS2144U=-3 or YAS2146=-3 or YAS2174=-3 or YAS2176=-3 then pvranch=-3;

/* PRESENT VALUE of mobile home/site. */

pvmbst=0; /* If the youth OWNS the mobile home and the site */
if YAS1500=1 then do;
    if (YAS2520=1 or YAS2520=2) and YAS2550=1 then pvmbst=YAS2552;
    if (YAS2520=1 or YAS2520=2) and YAS2550=2 then pvmbst=YAS2554L+(YAS2554U-YAS2554L)/2;
    if (YAS2520=1 or YAS2520=2) and ((YAS2550 ne 1 and YAS2550 ne 2) or (YAS2552=-1 or YAS2552=-2))
then do;
        if YAS2556=1 then do; pvmbst=500; flag=1; end;
        if YAS2556=2 then do; pvmbst=1750; flag=1; end;
        if YAS2556=3 then do; pvmbst=3750; flag=1; end;
        if YAS2556=4 then do; pvmbst=7500; flag=1; end;
        if YAS2556=5 then do; pvmbst=17500; flag=1; end;
        if YAS2556=6 then do; pvmbst=37500; flag=1; end;
        if YAS2556=7 then do; pvmbst=50001; flag=1; end;
    end;
end;
if YAS1500=-1 or YAS2520=-1 or YAS2554L=-1 or YAS2554U=-1 or YAS2556=-1 then pvmbst=-1;
if YAS1500=-2 or YAS2520=-2 or YAS2554L=-2 or YAS2554U=-2 or YAS2556=-2 then pvmbst=-2;
if YAS1500=-3 or YAS2520=-3 or YAS2552=-3 or YAS2554L=-3 or YAS2554U=-3 or YAS2556=-3 then pvmbst=-3;

pvmb=0; /* If the youth owns only the mobile home */
if YAS1500=1 then do;
    if (YAS2520=3 or YAS2520=4) and YAS2760=1 then pvmb=YAS2762;
    if (YAS2520=3 or YAS2520=4) and YAS2760=2 then pvmb=(YAS2764L+(YAS2764U-YAS2764L)/2);
    if (YAS2520=3 or YAS2520=4) and ((YAS2760 ne 1 and YAS2760 ne 2) or (YAS2762=-1 or YAS2762=-2))
then do;
        if YAS2766=1 then do; pvmb=12500; flag=1; end;
        if YAS2766=2 then do; pvmb=37500; flag=1; end;
        if YAS2766=3 then do; pvmb=75000; flag=1; end;
        if YAS2766=4 then do; pvmb=175000; flag=1; end;
        if YAS2766=5 then do; pvmb=375000; flag=1; end;
        if YAS2766=6 then do; pvmb=750000; flag=1; end;
    end;
end;

```

```

        if YAS2766=7 then do; pvmb=1000001; flag=1; end;
      end;
    end;

if YAS1500=-1 or YAS2520=-1 or YAS2764L=-1 or YAS2764U=-1 or YAS2766=-1 then pvmb=-1;
if YAS1500=-2 or YAS2520=-2 or YAS2764L=-2 or YAS2764U=-2 or YAS2766=-2 then pvmb=-2;
if YAS1500=-3 or YAS2520=-3 or YAS2762=-3 or YAS2764L=-3 or YAS2764U=-3 or YAS2766=-3 then pvmb=-3;

pvst=0; /* If the youth owns only the mobile home site */
if YAS1500=1 then do;
  if (YAS2520=5 or YAS2520=6) and YAS3010=1 then pvst=YAS3012;
  if (YAS2520=5 or YAS2520=6) and YAS3010=2 then pvst=(YAS3014L+(YAS3014U-YAS3014L)/2);
  if (YAS2520=5 or YAS2520=6) and ((YAS3010 ne 1 and YAS3010 NE 2) or (YAS3012=-1 or YAS3012=-2)) then do;
    if YAS3016=1 then do; pvst=12500; flag=1; end;
    if YAS3016=2 then do; pvst=57500; flag=1; end;
    if YAS3016=3 then do; pvst=75000; flag=1; end;
    if YAS3016=4 then do; pvst=175000; flag=1; end;
    if YAS3016=5 then do; pvst=575000; flag=1; end;
    if YAS3016=6 then do; pvst=750000; flag=1; end;
    if YAS3016=7 then do; pvst=1000001; flag=1; end;
  end;
end;

if YAS2520=-1 or YAS3014L=-1 or YAS3014U=-1 or YAS3016=-1 or YAS1500=-1 then pvst=-1;
if YAS2520=-2 or YAS3014L=-2 or YAS3014U=-2 or YAS3016=-2 or YAS1500=-2 then pvst=-2;
if YAS2520=-3 or YAS3012=-3 or YAS3014L=-3 or YAS3014U=-3 or YAS3016=-3 or YAS1500=-3 then pvst=-3;

/* If the respondent owns the apartment, its value */

pvapthm=0;
if YAS3310=1 or YAS3310=2 then do;
  if YAS3380=1 then pvapthm=YAS3382;
  if YAS3380=2 then pvapthm=(YAS3384L+(YAS3384U-YAS3384L)/2);
  if ((YAS3380 ne 1 and YAS3380 ne 2) or (YAS3382=-1 or YAS3382=-2)) then do;
    if YAS3386=1 then do; pvapthm=500; flag=1; end;
    if YAS3386=2 then do; pvapthm=1750; flag=1; end;
    if YAS3386=3 then do; pvapthm=3750; flag=1; end;
    if YAS3386=4 then do; pvapthm=7500; flag=1; end;
    if YAS3386=5 then do; pvapthm=17500; flag=1; end;
    if YAS3386=6 then do; pvapthm=37500; flag=1; end;
    if YAS3386=7 then do; pvapthm=50001; flag=1; end;
  end;
end;

if YAS3310=-1 or YAS3384L=-1 or YAS3384U=-1 or YAS3386=-1 then pvapthm=-1;
if YAS3310=-2 or YAS3384L=-2 or YAS3384U=-2 or YAS3386=-2 then pvapthm=-2;
if YAS3310=-3 or YAS3382=-3 or YAS3384L=-3 or YAS3384U=-3 or YAS3386=-3 then pvapthm=-3;

/* Any mortgage or land contract on land or property */

mortgagY=0;
if YAS3740=1 or YAS3740=2 then do;
  if YAS3750=1 and YAS3752 ge 0 then mortgagY=YAS3752;
  if YAS3750=2 and YAS3754U ge 0 and YAS3754L ge 0 then mortgagY=(YAS3754L+(YAS3754U-YAS3754L)/2);
  if ((YAS3750 ne 1 and YAS3750 NE 2) or (YAS3752=-1 or YAS3752=-2)) then do;

```

```

if YAS3756=1 then do; mortgagY=500; flag=1; end;
if YAS3756=2 then do; mortgagY=1750; flag=1; end;
if YAS3756=3 then do; mortgagY=3750; flag=1; end;
if YAS3756=4 then do; mortgagY=7500; flag=1; end;
if YAS3756=5 then do; mortgagY=17500; flag=1; end;
if YAS3756=6 then do; mortgagY=37500; flag=1; end;
if YAS3756=7 then do; mortgagY=50001; flag=1; end;
end;
end;

if YAS3740=-1 or YAS3754L=-1 or YAS3754U=-1 or YAS3756=-1 then mortgagY=-1;
if YAS3740=-2 or YAS3754L=-2 or YAS3754U=-2 or YAS3756=-2 then mortgagY=-2;
if YAS3740=-3 or YAS3752=-3 or YAS3754L=-3 or YAS3754U=-3 or YAS3756=-3 then mortgagY=-3;

/* Any loans to remodel this residence obtained in this second round */

loanowed=0;
if YAS3790=1 then do;
  if YAS3810=1 and YAS3812 ge 0 then loanowed=YAS3812;
  if YAS3810=2 and YAS3814U ge 0 and YAS3814L ge 0 then loanowed=(YAS3814L+(YAS3814U-
  YAS3814L)/2);
  if ((YAS3810 ne 1 and YAS3810 ne 2) or (YAS3812=-1 or YAS3812=-2)) then do;
    if YAS3816=1 then do; loanowed=12500; flag=1; end;
    if YAS3816=2 then do; loanowed=37500; flag=1; end;
    if YAS3816=3 then do; loanowed=75000; flag=1; end;
    if YAS3816=4 then do; loanowed=175000; flag=1; end;
    if YAS3816=5 then do; loanowed=375000; flag=1; end;
    if YAS3816=6 then do; loanowed=750000; flag=1; end;
    if YAS3816=7 then do; loanowed=1000001; flag=1; end;
  end;
end;

if YAS3780=-1 or YAS3785=-1 or YAS3790=-1 or YAS3814L=-1 or YAS3814U=-1 or YAS3816=-1 then
loanowed=-1;
if YAS3780=-2 or YAS3785=-2 or YAS3790=-2 or YAS3814L=-2 or YAS3814U=-2 or YAS3816=-2 then
loanowed=-2;
if YAS3780=-3 or YAS3785=-3 or YAS3790=-3 or YAS3812=-3 or YAS3814L=-3 or YAS3814U=-3 or YAS3816=-3
then loanowed=-3;

/* Any remaining loans from round one still unpaid, total amount owed */

stilowed=0;
if (YAS3840=1 and YAS3850=1) or YAS3860=1 then do;
  if YAS3880=1 and YAS3882 ge 0 then stilowed=YAS3882;
  if YAS3880=2 and YAS3884U ge 0 and YAS3884L ge 0 then stilowed=(YAS3884L+(YAS3884U-
  YAS3884L)/2);
  if ((YAS3880 ne 1 and YAS3880 ne 2) or (YAS3882=-1 or YAS3882=-2)) then do;
    if YAS3886=1 then do; stilowed=500; flag=1; end;
    if YAS3886=2 then do; stilowed=1750; flag=1; end;
    if YAS3886=3 then do; stilowed=3750; flag=1; end;
    if YAS3886=4 then do; stilowed=7500; flag=1; end;
    if YAS3886=5 then do; stilowed=17500; flag=1; end;
    if YAS3886=6 then do; stilowed=37500; flag=1; end;
    if YAS3886=7 then do; stilowed=50001; flag=1; end;
  end;
end;

```

```
if YAS3860=-1 or YAS3884L=-1 or YAS3884U=-1 or YAS3886=-1 then stilowed=-1;
if YAS3860=-2 or YAS3884L=-2 or YAS3884U=-2 or YAS3886=-2 then stilowed=-2;
if YAS3860=-3 or YAS3882=-3 or YAS3884L=-3 or YAS3884U=-3 or YAS3886=-3 then stilowed=-3;
```

```
/* Any second mortgages */
```

```
secmortY=0;
if YAS3910=1 then do;
    if YAS3920=1 and YAS3922 ge 0 then secmortY=YAS3922;
    if YAS3920=2 and YAS3924U ge 0 and YAS3924L ge 0 then secmortY=(YAS3924L+(YAS3924U-YAS3924L)/2);
    if ((YAS3920 ne 1 and YAS3920 ne 2) or (YAS3922=-1 or YAS3922=-2)) then do;
        if YAS3926=1 then do; secmortY=500; flag=1; end;
        if YAS3926=2 then do; secmortY=1750; flag=1; end;
        if YAS3926=3 then do; secmortY=3750; flag=1; end;
        if YAS3926=4 then do; secmortY=7500; flag=1; end;
        if YAS3926=5 then do; secmortY=17500; flag=1; end;
        if YAS3926=6 then do; secmortY=37500; flag=1; end;
        if YAS3926=7 then do; secmortY=50001; flag=1; end;
    end;
end;
```

```
if YAS3910=-1 or YAS3924L=-1 or YAS3924U=-1 or YAS3926=-1 then secmortY=-1;
if YAS3910=-2 or YAS3924L=-2 or YAS3924U=-2 or YAS3926=-2 then secmortY=-2;
if YAS3910=-3 or YAS3922=-3 or YAS3924L=-3 or YAS3924U=-3 or YAS3926=-3 then secmortY=-3;
```

```
/* Any taxes on the property to be paid */
```

```
proptaxY=0;
if YAS3950=1 then do;
    if YAS3960=1 and YAS3962 ge 0 then proptaxY=YAS3962;
    if YAS3960=2 and YAS3964U ge 0 and YAS3964L ge 0 then proptaxY=(YAS3964L+(YAS3964U-YAS3964L)/2);
    if ((YAS3960 ne 1 and YAS3960 ne 2) or (YAS3962=-1 or YAS3962=-2)) then do;
        if YAS3966=1 then do; proptaxY=500; flag=1; end;
        if YAS3966=2 then do; proptaxY=1750; flag=1; end;
        if YAS3966=3 then do; proptaxY=3750; flag=1; end;
        if YAS3966=4 then do; proptaxY=7500; flag=1; end;
        if YAS3966=5 then do; proptaxY=17500; flag=1; end;
        if YAS3966=6 then do; proptaxY=37500; flag=1; end;
        if YAS3966=7 then do; proptaxY=50001; flag=1; end;
    end;
end;
```

```
if YAS3950=-1 or YAS3964L=-1 or YAS3964U=-1 or YAS3966=-1 then proptaxY=-1;
if YAS3950=-2 or YAS3964L=-2 or YAS3964U=-2 or YAS3966=-2 then proptaxY=-2;
if YAS3950=-3 or YAS3962=-3 or YAS3964L=-3 or YAS3964U=-3 or YAS3966=-3 then proptaxY=-3;
```

```
/* Own a business, partnership or professional practice */
```

```
pvbussY=0;
if YAS4010=1 or YAS4010=2 then do;
    if YAS4030=1 then pbuss=YAS4032;
    if YAS4030=2 then pbuss=(YAS4034L+(YAS4034U-YAS4034L)/2);
    if ((YAS4030 ne 1 and YAS4030 ne 2) or (YAS4032=-1 or YAS4032=-2)) then do;
        if YAS4036=1 then do; pbuss=12500; flag=1; end;
        if YAS4036=2 then do; pbuss=37500; flag=1; end;
```

```

if YAS4036=3 then do; pbuss=75000; flag=1; end;
if YAS4036=4 then do; pbuss=175000; flag=1; end;
if YAS4036=5 then do; pbuss=375000; flag=1; end;
if YAS4036=6 then do; pbuss=750000; flag=1; end;
if YAS4036=7 then do; pbuss=1000001; flag=1; end;
end;
end;

if YAS4010=-1 or YAS4034L=-1 or YAS4034U=-1 or YAS4036=-1 then pbuss=-1;
if YAS4010=-2 or YAS4034L=-2 or YAS4034U=-2 or YAS4036=-2 then pbuss=-2;
if YAS4010=-3 or YAS4032=-3 or YAS4034L=-3 or YAS4034U=-3 or YAS4036=-3 then pbuss=-3;

/* Second real estate owned */

secrestY=0;
if YAS4140=1 or YAS4140=2 then do;
  if YAS4160=1 then secrestY=YAS4162;
  if YAS4160=2 then secrestY=(YAS4164L+(YAS4164U-YAS4164L)/2);
  if ((YAS4160 ne 1 and YAS4160 ne 2) or (YAS4162=-1 or YAS4162=-2)) then do;
    if YAS4166=1 then do; secrestY=12500; flag=1; end;
    if YAS4166=2 then do; secrestY=37500; flag=1; end;
    if YAS4166=3 then do; secrestY=75000; flag=1; end;
    if YAS4166=4 then do; secrestY=175000; flag=1; end;
    if YAS4166=5 then do; secrestY=375000; flag=1; end;
    if YAS4166=6 then do; secrestY=750000; flag=1; end;
    if YAS4166=7 then do; secrestY=1000001; flag=1; end;
  end;
end;

if YAS4140=-1 or YAS4164L=-1 or YAS4164U=-1 or YAS4166=-1 then secrestY=-1;
if YAS4140=-2 or YAS4164L=-2 or YAS4164U=-2 or YAS4166=-2 then secrestY=-2;
if YAS4140=-3 or YAS4162=-3 or YAS4164L=-3 or YAS4164U=-3 or YAS4166=-3 then secrestY=-3;

/* Any retirement plans or pensions */

retireY=0;
if YAS4270=1 or YAS4270=2 or YAS4270=3 then do;
  if YAS4290=1 and YAS4292 ge 0 then retireY=YAS4292;
  if YAS4290=2 and YAS4294U ge 0 and YAS4294L ge 0 then retireY=(YAS4294L+(YAS4294U-YAS4294L)/2);
  if ((YAS4290 ne 1 and YAS4290 ne 2) or (YAS4292=-1 or YAS4292=-2)) then do;
    if YAS4296=1 then do; retireY=2500; flag=1; end;
    if YAS4296=2 then do; retireY=7500; flag=1; end;
    if YAS4296=3 then do; retireY=17500; flag=1; end;
    if YAS4296=4 then do; retireY=37500; flag=1; end;
    if YAS4296=5 then do; retireY=75000; flag=1; end;
    if YAS4296=6 then do; retireY=175000; flag=1; end;
    if YAS4296=7 then do; retireY=250001; flag=1; end;
  end;
end;

if YAS4270=-1 or YAS4294L=-1 or YAS4294U=-1 or YAS4296=-1 then retireY=-1;
if YAS4270=-2 or YAS4294L=-2 or YAS4294U=-2 or YAS4296=-2 then retireY=-2;
if YAS4270=-3 or YAS4292=-3 or YAS4294L=-3 or YAS4294U=-3 or YAS4296=-3 then retireY=-3;

/* Any savings in saving accounts, money market, funds, trusts,...*/

savingsY=0;

```

```

if YAS4400=1 or YAS4400=2 or YAS4400=3 then do;
    if YAS4420=1 and YAS4422 ge 0 then savingsY=YAS4422;
    if YAS4420=2 and YAS4424U ge 0 and YAS4424L ge 0 then savingsY=(YAS4424L+(YAS4424U-
YAS4424L)/2);
    if ((YAS4420 ne 1 and YAS4420 ne 2) or (YAS4422=-1 or YAS4422=-2)) then do;
        if YAS4426=1 then do; savingsY=500; flag=1; end;
        if YAS4426=2 then do; savingsY=1750; flag=1; end;
        if YAS4426=3 then do; savingsY=3750; flag=1; end;
        if YAS4426=4 then do; savingsY=7500; flag=1; end;
        if YAS4426=5 then do; savingsY=17500; flag=1; end;
        if YAS4426=6 then do; savingsY=37500; flag=1; end;
        if YAS4426=7 then do; savingsY=50001; flag=1; end;
    end;
end;

if YAS4400=-1 or YAS4424L=-1 or YAS4424U=-2 or YAS4426=-1 then savingsY=-1;
if YAS4400=-2 or YAS4424L=-2 or YAS4424U=-2 or YAS4426=-2 then savingsY=-2;
if YAS4400=-3 or YAS4422=-3 or YAS4424L=-3 or YAS4424U=-3 or YAS4426=-3 then savingsY=-3;

/* Any other savings in bonds or CDs */

othsavY=0;
if YAS4530=1 or YAS4530=2 or YAS4530=3 then do;
    if YAS4550=1 and YAS4552 ge 0 then othsavY=YAS4552;
    if YAS4550=2 and YAS4554U ge 0 and YAS4554L ge 0 then othsavY=(YAS4554L+(YAS4554U-
YAS4554L)/2);
    if ((YAS4550 ne 1 and YAS4550 ne 2) or (YAS4552=-1 or YAS4552=-2)) then do;
        if YAS4556=1 then do; othsavY=500; flag=1; end;
        if YAS4556=2 then do; othsavY=1750; flag=1; end;
        if YAS4556=3 then do; othsavY=3750; flag=1; end;
        if YAS4556=4 then do; othsavY=7500; flag=1; end;
        if YAS4556=5 then do; othsavY=17500; flag=1; end;
        if YAS4556=6 then do; othsavY=37500; flag=1; end;
        if YAS4556=7 then do; othsavY=50001; flag=1; end;
    end;
end;

if YAS4530=-1 or YAS4554L=-1 or YAS4554U=-1 or YAS4556=-1 then othsavY=-1;
if YAS4530=-2 or YAS4554L=-2 or YAS4554U=-2 or YAS4556=-2 then othsavY=-2;
if YAS4530=-3 or YAS4552=-3 or YAS4554L=-3 or YAS4554U=-3 or YAS4556=-3 then othsavY=-3;

/* Any stocks in corporations, mutual funds */

stockY=0;
if YAS4660=1 or YAS4660=2 or YAS4660=3 then do;
    if YAS4680=1 then stockY=YAS4682;
    if YAS4680=2 then stockY=(YAS4684L+(YAS4684U-YAS4684L)/2);
    if ((YAS4680 ne 1 and YAS4680 ne 2) or (YAS4682=-1 or YAS4682=-2)) then do;
        if YAS4686=1 then do; stockY=500; flag=1; end;
        if YAS4686=2 then do; stockY=1750; flag=1; end;
        if YAS4686=3 then do; stockY=3750; flag=1; end;
        if YAS4686=4 then do; stockY=7500; flag=1; end;
        if YAS4686=5 then do; stockY=17500; flag=1; end;
        if YAS4686=6 then do; stockY=37500; flag=1; end;
        if YAS4686=7 then do; stockY=50001; flag=1; end;
    end;
end;

```

```

if YAS4660=-1 or YAS4684L=-1 or YAS4684U=-1 or YAS4686=-1 then stockY=-1;
if YAS4660=-2 or YAS4684L=-2 or YAS4684U=-2 or YAS4686=-2 then stockY=-2;
if YAS4660=-3 or YAS4682=-3 or YAS4684L=-3 or YAS4684U=-3 or YAS4686=-3 then stockY=-3;

```

```

/* Present value of any vehicles owned */

```

```

pvcarsY=0;
if YAS4790=1 or YAS4790=2 or YAS4790=3 then do;
    if YAS4810=1 then pvcarsY=YAS4812;
    if YAS4810=2 then pvcarsY=(YAS4814L+(YAS4814U-YAS4814L)/2);
    if ((YAS4810 ne 1 and YAS4810 ne 2) or (YAS4812=-1 or YAS4812=-2)) then do;
        if YAS4816=1 then do; pvcarsY=2500; flag=1; end;
        if YAS4816=2 then do; pvcarsY=7500; flag=1; end;
        if YAS4816=3 then do; pvcarsY=17500; flag=1; end;
        if YAS4816=4 then do; pvcarsY=37500; flag=1; end;
        if YAS4816=5 then do; pvcarsY=75000; flag=1; end;
        if YAS4816=6 then do; pvcarsY=175000; flag=1; end;
        if YAS4816=7 then do; pvcarsY=250001; flag=1; end;
    end;
end;

```

```

if YAS4790=-1 or YAS4814L=-1 or YAS4814U=-1 or YAS4816=-1 then pvcarsY=-1;
if YAS4790=-2 or YAS4814L=-2 or YAS4814U=-2 or YAS4816=-2 then pvcarsY=-2;
if YAS4790=-3 or YAS4812=-3 or YAS4814L=-3 or YAS4814U=-3 or YAS4376=-3 then pvcarsY=-3;

```

```

/* Money still owed on these vehicles */

```

```

cardebt=0;
if YAS4840=1 then cardebt=YAS4842;
if YAS4840=2 then cardebt=(YAS4844L+(YAS4844U-YAS4844L)/2);
if ((YAS4840 ne 1 and YAS4840 ne 2) or (YAS4842=-1 or YAS4842=-2)) then do;
    if YAS4846=1 then do; cardebt=2500; flag=1; end;
    if YAS4846=2 then do; cardebt=7500; flag=1; end;
    if YAS4846=3 then do; cardebt=17500; flag=1; end;
    if YAS4846=4 then do; cardebt=37500; flag=1; end;
    if YAS4846=5 then do; cardebt=75000; flag=1; end;
    if YAS4846=6 then do; cardebt=175000; flag=1; end;
    if YAS4846=7 then do; cardebt=250001; flag=1; end;
end;

```

```

if YAS4790=-1 or YAS4844L=-1 or YAS4844U=-1 or YAS4846=-1 then cardebt=-1;
if YAS4790=-2 or YAS4844L=-2 or YAS4844U=-2 or YAS4846=-2 then cardebt=-2;
if YAS4790=-3 oR YAS4842=-3 or YAS4844L=-3 or YAS4844U=-3 or YAS4376=-3 then cardebt=-3;

```

```

/* Present value of owned furniture */

```

```

pvfurnty=0;
if YAS4870=1 then do; pvfurnty=2500; flag=1; end;
if YAS4870=2 then do; pvfurnty=7500; flag=1; end;
if YAS4870=3 then do; pvfurnty=17500; flag=1; end;
if YAS4870=4 then do; pvfurnty=37500; flag=1; end;
if YAS4870=5 then do; pvfurnty=75000; flag=1; end;
if YAS4870=6 then do; pvfurnty=175000; flag=1; end;
if YAS4870=7 then do; pvfurnty=250001; flag=1; end;

```

```

if YAS4870=-1 then pvfurnty=-1;

```

```

if YAS4870=-2 then pvfurnty=-2;
if YAS4870=-3 then pvfurnty=-3;

/* Any other assets not being mentioned before */

otassetY=0;
if YAS4880=1 or YAS4880=2 or YAS4880=3 then do;
  if YAS4900=1 then otassetY=YAS4902;
  if YAS4900=2 then otassetY=(YAS4904L+(YAS4904U-YAS4904L)/2);
  if ((YAS4900 ne 1 and YAS4900 ne 2) or (YAS4902=-1 or YAS4902=-2)) then do;
    if YAS4906=1 then do; otassetY=2500; flag=1; end;
    if YAS4906=2 then do; otassetY=7500; flag=1; end;
    if YAS4906=3 then do; otassetY=17500; flag=1; end;
    if YAS4906=4 then do; otassetY=37500; flag=1; end;
    if YAS4906=5 then do; otassetY=75000; flag=1; end;
    if YAS4906=6 then do; otassetY=175000; flag=1; end;
    if YAS4906=7 then do; otassetY=250001; flag=1; end;
  end;
end;

if YAS4880=-1 or YAS4904L=-1 or YAS4904U=-1 or YAS4906=-1 then otassetY=-1;
if YAS4880=-2 or YAS4904L=-2 or YAS4904U=-2 or YAS4906=-2 then otassetY=-2;
if YAS4880=-3 or YAS4902=-3 or YAS4904L=-3 or YAS4904U=-3 or YAS4376=-3 then otassetY=-3;

/* Any loans still owed to family or relatives */

array rloan rloan01 rloan02 rloan03 rloan04 rloan05 rloan06 rloan07 rloan08 rloan09 rloan10 rloan11 rloan12
      rloan13 rloan14 rloan15 rloan16 rloan17 rloan18 rloan19 rloan20;
array A5080 A508001 A508002 A508003 A508004 A508005 A508006 A508007 A508008 A508009 A508010
      A508011 A508012 A508013 A508014 A508015 A508016 A508017 A508018 A508019 A508020;
array A5082 A508201 A508202 A508203 A508204 A508205 A508206 A508207 A508208 A508209 A508210
      A508211 A508212 A508213 A508214 A508215 A508216 A508217 A508218 A508219 A508220;
array A5084L A5084L01 A5084L02 A5084L03 A5084L04 A5084L05 A5084L06 A5084L07 A5084L08 A5084L09
      A5084L10 A5084L11 A5084L12 A5084L13 A5084L14 A5084L15 A5084L16 A5084L17 A5084L18
      A5084L19 A5084L20;
array A5084U A5084U01 A5084U02 A5084U03 A5084U04 A5084U05 A5084U06 A5084U07 A5084U08
      A5084U09 A5084U10 A5084U11 A5084U12 A5084U13 A5084U14 A5084U15 A5084U16 A5084U17
      A5084U18 A5084U19 A5084U20;
array A5086 A508601 A508602 A508603 A508604 A508605 A508606 A508607 A508608 A508609 A508610
      A508611 A508612 A508613 A508614 A508615 A508616 A508617 A508618 A508619 A508620;
array A5170 A517001 A517002 A517003 A517004 A517005 A517006 A517007 A517008 A517009 A517010
      A517011 A517012 A517013 A517014 A517015 A517016 A517017 A517018 A517019 A517020;
array A5172 A517201 A517202 A517203 A517204 A517205 A517206 A517207 A517208 A517209 A517210
      A517211 A517212 A517213 A517214 A517215 A517216 A517217 A517218 A517219 A517220;
array A5174L A5174L01 A5174L02 A5174L03 A5174L04 A5174L05 A5174L06 A5174L07 A5174L08 A5174L09
      A5174L10 A5174L11 A5174L12 A5174L13 A5174L14 A5174L15 A5174L16 A5174L17 A5174L18
      A5174L19 A5174L20;
array A5174U A5174U01 A5174U02 A5174U03 A5174U04 A5174U05 A5174U06 A5174U07 A5174U08
      A5174U09 A5174U10 A5174U11 A5174U12 A5174U13 A5174U14 A5174U15 A5174U16 A5174U17
      A5174U18 A5174U19 A5174U20;
array A5176 A517601 A517602 A517603 A517604 A517605 A517606 A517607 A517608 A517609 A517610
      A517611 A517612 A517613 A517614 A517615 A517616 A517617 A517618 A517619 A517620;
array A5074 A507401-A507420;
array A5066 A506601-A506620;
array A5164 A516401-A516420;

do I=1 to 20;

```

```

rloan(I)=0;
if YAS5040=1 and YAS5060 ne 1 and YAS5062 ne 1 then do;
    if A5080(I)=1 then rloan(I)=A5082(I);
    if A5080(I)=2 then rloan(I)=(A5084L(I)+(A5084U(I)-A5084L(I))/2);
    if ((A5080(I) ne 1 and A5080(I) ne 2) or (A5082(I)=-1 or A5082(I)=-2)) then do;
        if A5086(I)=1 then do; rloan(I)=500; flag=1; end;
        if A5086(I)=2 then do; rloan(I)=1750; flag=1; end;
        if A5086(I)=3 then do; rloan(I)=3750; flag=1; end;
        if A5086(I)=4 then do; rloan(I)=7500; flag=1; end;
        if A5086(I)=5 then do; rloan(I)=17500; flag=1; end;
        if A5086(I)=6 then do; rloan(I)=37500; flag=1; end;
        if A5086(I)=7 then do; rloan(I)=50001; flag=1; end;
    end;
end;

if YAS5040=-1 or A5074(I)=-1 or A5086(I)=-1 or A5084L[I]=-1 or A5084U[I]=-1 then rloan(I)=-1;
if YAS5040=-2 OR A5074(I)=-2 or A5086(I)=-2 or A5084L[I]=-2 or A5084U[I]=-2 then rloan(I)=-2;
if YAS5040=-3 or A5082(I)=-3 or A5086(I)=-3 or A5084L[I]=-3 or A5084U[I]=-3 then rloan(I)=-3;

if YAS5130=1 and YAS5150 ne 1 and YAS5152 ne 1 then do;
    if A5170(I)=1 then rloan(I)=A5172(I);
    if A5170(I)=2 then rloan(I)=(A5174L(I)+(A5174U(I)-A5174L(I))/2);
    if ((A5170(I) ne 1 and A5170(I) ne 2) or (A5172(I)=-1 or A5172(I)=-2)) then do;
        if A5176(I)=1 then do; rloan(I)=500; flag=1; end;
        if A5176(I)=2 then do; rloan(I)=1750; flag=1; end;
        if A5176(I)=3 then do; rloan(I)=3750; flag=1; end;
        if A5176(I)=4 then do; rloan(I)=7500; flag=1; end;
        if A5176(I)=5 then do; rloan(I)=17500; flag=1; end;
        if A5176(I)=6 then do; rloan(I)=37500; flag=1; end;
        if A5176(I)=7 then do; rloan(I)=50001; flag=1; end;
    end;
end;

if YAS5130=-1 or A5164(I)=-1 or A5176(I)=-1 or A5174L[I]=-1 or A5174U[I]=-1 then rloan(I)=-1;
if YAS5130=-2 or A5164(I)=-2 or A5176(I)=-2 or A5174L[I]=-2 or A5174U[I]=-2 then rloan(I)=-2;
if YAS5130=-3 or A5172(I)=-3 or A5176(I)=-3 or A5174L[I]=-3 or A5174U[I]=-3 then rloan(I)=-3;

end;

/* Any other debts from loans, credit cards, etc...*/

othdebtY=0;
if YAS5210=1 then do;
    if YAS5220=1 then othdebtY=YAS5222;
    if YAS5220=2 then othdebtY=(YAS5224L+(YAS5224U-YAS5224L)/2);
    if ((YAS5220 ne 1 and YAS5220 ne 2) or (YAS5222=-1 or YAS5222=-2)) then do;
        if YAS5226=1 then do; othdebtY=500; flag=1; end;
        if YAS5226=2 then do; othdebtY=1750; flag=1; end;
        if YAS5226=3 then do; othdebtY=3750; flag=1; end;
        if YAS5226=4 then do; othdebtY=7500; flag=1; end;
        if YAS5226=5 then do; othdebtY=17500; flag=1; end;
        if YAS5226=6 then do; othdebtY=37500; flag=1; end;
        if YAS5226=7 then do; othdebtY=50001; flag=1; end;
    end;
end;

if YAS5210=-1 or YAS5224L=-1 or YAS5224U=-1 or YAS5226=-1 then othdebtY=-1;

```

```

if YAS5210=-2 or YAS5224L=-2 or YAS5224U=-2 or YAS5226=-2 then othdebtY=-2;
if YAS5210=-3 or YAS5222=-3 or YAS5224L=-3 or YAS5224U=-3 or YAS5226=-3 then othdebtY=-3;

/* We now calculate the household net worth according to the youth: hhworthY=assets-liabilities */

hhworthY=0;
if pvranch not in (-1, -2, -3) then hhworthY=hhworthY+pvranch;
if pvmbst not in (-1, -2, -3) then hhworthY=hhworthY+pvmbst;
if pvmb not in (-1, -2, -3) then hhworthY=hhworthY+pvmb;
if pvst not in (-1, -2, -3) then hhworthY=hhworthY+pvst;
if pvapthm not in (-1, -2, -3) then hhworthY=hhworthY+pvapthm;
if pvbussY not in (-1, -2, -3) then hhworthY=hhworthY+pvbussY;
if secrestY not in (-1, -2, -3) then hhworthY=hhworthY+secrestY;
if retireY not in (-1, -2, -3) then hhworthY=hhworthY+retireY;
if savingsY not in (-1, -2, -3) then hhworthY=hhworthY+savingsY;
if othsavY not in (-1, -2, -3) then hhworthY=hhworthY+othsavY;
if stockY not in (-1, -2, -3) then hhworthY=hhworthY+stockY;
if pvcarsY not in (-1, -2, -3) then hhworthY=hhworthY+pvcarsY;
if pfurnty not in (-1, -2, -3) then hhworthY=hhworthY+pfurnty;
if otassetY not in (-1, -2, -3) then hhworthY=hhworthY+otassetY;
if mortgagY not in (-1, -2, -3) then hhworthY=hhworthY-mortgagY;
if loanowed not in (-1, -2, -3) then hhworthY=hhworthY-loanowed;
if stilowed not in (-1, -2, -3) then hhworthY=hhworthY-stilowed;
if secmortY not in (-1, -2, -3) then hhworthY=hhworthY-secmortY;
if cardebt not in (-1, -2, -3) then hhworthY=hhworthY-cardebt;
if proptaxY not in (-1, -2, -3) then hhworthY=hhworthY-proptaxY;
if othdebtY not in (-1, -2, -3) then hhworthY=hhworthY-othdebtY;
do I=1 to 19;
    if rloan[I] not in (-1, -2, -3) then hhworthY=hhworthY-rloan[I];
end;
end; /* CORRESPONDING TO YAS50=1 */

if YAS50=0 or YAS50=-4 then hhworthY=-4;
if YAS50=-1 or (pvranch=-1 or pvmbast=-1 or pvmbhm=-1 or pvapthm=-1 or pvbussY=-1 or
    secrestY=-1 or retireY=-1 or savingsY=-1 or othsavY=-1 or stockY=-1 or pvcarsY=-1 or pfurnty=-1 or otassetY=-1 or
    mortgagY=-1 or loanowed=-1 or stilowed=-1 or secmortY=-1 or cardebt=-1 or proptaxY=-1 or rloan01=-1 or
    rloan02=-1 or rloan03=-1 or rloan04=-1 or rloan05=-1 or rloan06=-1 or rloan07=-1 or rloan08=-1 or
    rloan09=-1 or rloan10=-1 or rloan11=-1 or rloan12=-1 or rloan13=-1 or rloan14=-1 or rloan15=-1 or rloan16=-1 or
    rloan17=-1 or rloan18=-1 or rloan19=-1 or othdebtY=-1)
then hhworthY=-1;
if YAS50=-2 or (pvranch=-2 or pvmbast=-2 or pvmbhm=-2 or pvapthm=-2 or pvbussY=-2 or
    secrestY=-2 or retireY=-2 or savingsY=-2 or othsavY=-2 or stockY=-2 or pvcarsY=-2 or pfurnty=-2 or otassetY=-2 or
    mortgagY=-2 or loanowed=-2 or stilowed=-2 or secmortY=-2 or cardebt=-2 or proptaxY=-2 or rloan01=-2 or
    rloan02=-2 or rloan03=-2 or rloan04=-2 or rloan05=-2 or rloan06=-2 or rloan07=-2 or rloan08=-2 or
    rloan09=-2 or rloan10=-2 or rloan11=-2 or rloan12=-2 or rloan13=-2 or rloan14=-2 or rloan15=-2 or rloan16=-2 or
    rloan17=-2 or rloan18=-2 or rloan19=-2 or othdebtY=-2)
then hhworthY=-2;
if YAS50=-3 or (pvranch=-3 or pvmbast=-3 or pvmbhm=-3 or pvapthm=-3 or pvbussY=-3 or
    secrestY=-3 or retireY=-3 or savingsY=-3 or othsavY=-3 or stockY=-3 or pvcarsY=-3 or pfurnty=-3 or otassetY=-3 or
    mortgagY=-3 or loanowed=-3 or stilowed=-3 or secmortY=-3 or cardebt=-3 or proptaxY=-3 or rloan01=-3 or
    rloan02=-3 or rloan03=-3 or rloan04=-3 or rloan05=-3 or rloan06=-3 or rloan07=-3 or rloan08=-3 or
    rloan09=-3 or rloan10=-3 or rloan11=-3 or rloan12=-3 or rloan13=-3 or rloan14=-3 or rloan15=-3 or rloan16=-3 or
    rloan17=-3 or rloan18=-3 or rloan19=-3 or othdebtY=-3)
then hhworthY=-3;
if YAS50=-5 then hhworthY=-5; /*give the people who were not interviewed in round 2 value -5.*/

```

***** SECTION 3: HOUSEHOLD POVERTY STATUS *****

```

povert=-4;
if HHSIZE=-1 or under18=-1 then povert=-1;
if HHSIZE=-2 or under18=-2 then povert=-2;
if HHSIZE=-3 or under18=-3 then povert=-3;

if HHSIZE=1 then povert=8350;

if HHSIZE=2 then do;
    if under18=0 then povert=10748;
    if under18=1 then povert=11063;
end;

if HHSIZE=3 then do;
    if under18=0 then povert=12554;
    if under18=1 then povert=12919;
    if under18=2 then povert=12931;
end;

if HHSIZE=4 then do;
    if under18=0 then povert=16555;
    if under18=1 then povert=16825;
    if under18=2 then povert=16276;
    if under18=3 then povert=16333;
end;

if HHSIZE=5 then do;
    if under18=0 then povert=19964;
    if under18=1 then povert=20255;
    if under18=2 then povert=19634;
    if under18=3 then povert=19154;
    if under18=4 then povert=18861;
end;

if HHSIZE=6 then do;
    if under18=0 then povert=22962;
    if under18=1 then povert=23053;
    if under18=2 then povert=22578;
    if under18=3 then povert=22123;
    if under18=4 then povert=21446;
    if under18=5 then povert=21045;
end;

if HHSIZE=7 then do;
    if under18=0 then povert=26421;
    if under18=1 then povert=26586;
    if under18=2 then povert=26017;
    if under18=3 then povert=25621;
    if under18=4 then povert=24882;
    if under18=5 then povert=24021;
    if under18=6 then povert=23076;
end;

```

```

if HHSIZE=8 then do;
    if under18=0 then povert=29550;
    if under18=1 then povert=29811;
    if under18=2 then povert=29274;
    if under18=3 then povert=28804;
    if under18=4 then povert=28137;
    if under18=5 then povert=27290;
    if under18=6 then povert=26409;
    if under18=7 then povert=26185;
end;

if HHSIZE>=9 then do;
    if under18=0 then povert=35546;
    if under18=1 then povert=35719;
    if under18=2 then povert=35244;
    if under18=3 then povert=34845;
    if under18=4 then povert=34190;
    if under18=5 then povert=33289;
    if under18=6 then povert=32474;
    if under18=7 then povert=32272;
    if under18=8 then povert=31029;
end;

povthr=-4;
if groshhIY ge 0 and povert gt 0 then
    povthr=groshhIY/povert;
if groshhIY eq -1 or povert=-1 then povthr=-1;
if groshhIY eq -2 or povert=-2 then povthr=-2;
if groshhIY eq -3 or povert=-3 then povthr=-3;
if groshhIY eq -4 then povthr=-4;

povthr2=-4;
if povthr ge 0 then do;
    povthr2=povthr*100;
end;
if povthr eq -1 then povthr2=-1;
if povthr eq -2 then povthr2=-2;
if povthr eq -3 then povthr2=-3;

povthr3=-4;
if povthr2 ge 0 then do;
    povthr3=round(povthr2, 1);
end;
if povthr2 eq -1 then povthr3=-1;
if povthr2 eq -2 then povthr3=-2;
if povthr2 eq -3 then povthr3=-3;

/* Respondents not interviewed in round 2 */
if YAS50=-5 then povthr3=-5;

endsas;

```

PARTICIPATION IN GOVERNMENT PROGRAMS

Variables Created: CV_AMT_GOVNT_PGM_PCY.80 – CV_AMT_GOVNT_PGM_PCY.99
 CV_GOVNT_PGM_EVER
 CV_GOVNT_PGM_YR.80 – CV_GOVNT_PGM_YR.99

Variables Used

Name in Program	Question Name on CD	Name in Program	Question Name on CD
PUBID	PUBID	BDATE_D,_M,_Y	KEY!BDATE_D,_M,_Y (rd. 1)
P1210	YPRG-1210	LINT_D,LINT_M,LINT_Y	CV_INTERVIEW_DATE_D,_M,_Y (rd. 1)
P1895	YPRG-1895_R2	PS4001M, PS4001Y	YPRG-4000_R2.01~M, ~Y
P1905	YPRG-1905_R2	PS47001M, PS47001Y	YPRG-4700_R2.01~M, ~Y
P1920	YPRG-1920_R2	PS10201M, PS10201Y	YPRG-10200_R2.01~M, ~Y
P1930	YPRG-1930_R2	PS10901M, PS10901Y	YPRG-10900_R2.01~M, ~Y
P39011	YPRG-3901_R2.01	P16390M, P16390Y	YPRG-16390_R2~M, ~Y
PS42001	YPRG-4200_R2.01	P191001M, P191001Y	YPRG-19100_R2.01~M, ~Y
PS44001	YPRG-4400_R2.01	P191002M, P191002Y	YPRG-19100_R2.02~M, ~Y
PS49001	YPRG-4900_R2.01	PS20001M, PS20001Y	YPRG-20000_R2.01~M, ~Y
PS59001	YPRG-5900_R2.01	PS20002M, PS20002Y	YPRG-20000_R2.02~M, ~Y
PS61001	YPRG-6100_R2.01	PS209011-PS209015	YPRG-20900_R2.01~000001 - ~000005
P6200B1	YPRG-6200B_R2.01	P21410M, P21410Y	YPRG-21410_R2~M, ~Y
PS63001	YPRG-6300_R2.01	PS22001M, PS22001Y	YPRG-22000_R2.01~M, ~Y
P9865	YPRG-9865_R2	PS22002M, PS22002Y	YPRG-22000_R2.02~M, ~Y
P9867	YPRG-9867_R2	PS22601M, PS22601Y	YPRG-22600_R2.01~M, ~Y
P9890	YPRG-9890_R2	PS22602M, PS22602Y	YPRG-22600_R2.02~M, ~Y
P9900	YPRG-9900_R2	PS235011-PS235015	YPRG-23500_R2.01~000001 - ~000005
P10100A1	YPRG-10100A_R2.01	PS235199	YPRG-23500_R2.01~000099
PS104001	YPRG-10400_R2.01	PS235021-PS235025	YPRG-23500_R2.02~000001 - ~000005
PS106001	YPRG-10600_R2.01	P235299	YPRG-23500_R2.02~000099
PS111001	YPRG-11100_R2.01	P235031-P235035	YPRG-23500_R2.03~000001 - ~000005
PS121001	YPRG-12100_R2.01	P2350399	YPRG-23500_R2.03~000099
P12500A1	YPRG-12500A_R2.01	P358001M, P358001Y	YPRG-35800_R2.01~M, ~Y
PS126001	YPRG-12600_R2.01	P358002M, P358002Y	YPRG-35800_R2.02~M, ~Y
P16385	YPRG-16385_R2	P358003M, P358003Y	YPRG-35800_R2.03~M, ~Y
P16400	YPRG-16400_R2	P358401M, P358401Y	YPRG-35840_R2.01~M, ~Y
P16410	YPRG-16410_R2	P358402M, P358402Y	YPRG-35840_R2.02~M, ~Y
P19063A1, P19063A2	YPRG-19063A_R2.01, .02	P3592011-P3592015	YPRG-35920_R2.01~000001 - ~000005
PS202001	YPRG-20200_R2.01	P3592021-P3592025	YPRG-35920_R2.02~000001 - ~000005
PS207001, PS207002	YPRG-20700_R2.01	P3592031-P3592035	YPRG-35920_R2.03~000001 - ~000005
PS208001	YPRG-20800_R2.01	P36100M, P36100Y	YPRG-36100_R2~M, ~Y
P21100A1, P21100A2	YPRG-21100A_R2.01, .02	PS31001M, PS31001Y	YPRG-31000_R2.01~M, ~Y
P21397	YPRG-21397_R2	PS31002M, PS31002Y	YPRG-31000_R2.02~M, ~Y
P21405	YPRG-21405_R2	PS31601M, PS31601Y	YPRG-31600_R2.01~M, ~Y
P21420	YPRG-21420_R2	PS31602M, PS31602Y	YPRG-31600_R2.02~M, ~Y
P21430	YPRG-21430_R2	PS325011-PS325015	YPRG-32500_R2.01~000001 - ~000005
P21900A1-P21900A3	YPRG-21900A_R2.01-.03	PS325021-PS325025	YPRG-32500_R2.02~000001 - ~000005
PS228001, PS228002	YPRG-22800_R2.01, .02	P40001M, P40001Y	YPRG-4000.01~M, ~Y
PS233001-PS233003	YPRG-23300_R2.01-.03	P40002M, P40002Y	YPRG-4000.02~M, ~Y
PS234001, PS234002	YPRG-23400_R2.01, .02	P40003M, P40003Y	YPRG-4000.03~M, ~Y
P23700A1-P23700A3	YPRG-23700A_R2.01-.03	P47001M, P47001Y	YPRG-4700.01~M, ~Y
P35715	YPRG-35715_R2	P47002M, P47002Y	YPRG-4700.02~M, ~Y
P35730	YPRG-35730_R2	P47003M, P47003Y	YPRG-4700.03~M, ~Y
P35740	YPRG-35740_R2	P102001M, P102001Y	YPRG-10200.01~M, ~Y
P357901-P357903	YPRG-35790_R2.01-.03	P109001M, P109001Y	YPRG-10900.01~M, ~Y
P359001-P359003	YPRG-35900_R2.01-.03	P194001M, P194001Y	YPRG-19400.01~M, ~Y
P359101	YPRG-35910_R2.01	P194002M, P194002Y	YPRG-19400.02~M, ~Y
P359501-P359503	YPRG-35950_R2.01-.03	P194003M, P194003Y	YPRG-19400.03~M, ~Y
P36087	YPRG-36087_R2	P200001M, P200001Y	YPRG-20000.01~M, ~Y
P36095	YPRG-36095_R2	P200002M, P200002Y	YPRG-20000.02~M, ~Y
P36110	YPRG-36110_R2	P2090011-P2090015	YPRG-20900.01~000001 - ~000005
P36120	YPRG-36120_R2	P2090021-P2090025	YPRG-20900.02~000001 - ~000005

Appendix 5: Income and Assets Variable Creation

P30900A1, P30900A2	YPRG-30900A_R2.01, .02	P2090031-P2090035	YPRG-20900.03~000001 - ~000005
PS318001	YPRG-31800_R2.01	P220001M, P220001Y	YPRG-22000.01~M, ~Y
PS323001, PS323002	YPRG-32300_R2.01, .02	P220002M, P220002Y	YPRG-22000.02~M, ~Y
PS324001	YPRG-32400_R2.01	P226001M, P226001Y	YPRG-22600.01~M, ~Y
P32700A1 P32700A2	YPRG-32700A_R2.01, .02	P2350011-P2350015	YPRG-23500.01~000001 - ~000005
P1900	YPRG-1900	P2350021 P2350025	YPRG-23500.02~000001 - ~000005
P2000	YPRG-2000	P167001M, P167001Y	YPRG-16700.01~M, ~Y
P2500	YPRG-2500	P172001M, P172001Y	YPRG-17200.01~M, ~Y
P2600	YPRG-2600	P1830011-P1830015	YPRG-18300.01~000001 - ~000005
P2400	YPRG-2400	P310001M, P310001Y	YPRG-31000.01~M, ~Y
P3300	YPRG-3300	P310002M, P310002Y	YPRG-31000.02~M, ~Y
P3400	YPRG-3400	P316001M, P316001Y	YPRG-31600.01~M, ~Y
P3500	YPRG-3500	P3250011-P3250015	YPRG-32500.01~000001 - ~000005
P42001-P42003	YPRG-4200.01-.03	P3250021-P3250025	YPRG-32500.02~000001 - ~000005
P44001-P44003	YPRG-4400.01-.03	YINT_D, YINT_M, YINT_Y	YINTDATE-D, ~M, ~Y
P45001, P45002	YPRG-4500.01, .02	P18900	YPRG-18900
P48001-P48003	YPRG-4800.01-.03	P196001-P196003	YPRG-19600.01-.03
P49001, P49002	YPRG-4900.01, .02	P201001-P201003	YPRG-20100.01-.03
P52001-P52003	YPRG-5200.01-.03	P202001, P202002	YPRG-20200.01, .02
P56001-P56003	YPRG-5600.01-.03	P207001-P207003	YPRG-20700.01-.03
P59001-P59003	YPRG-5900.01-.03	P208001	YPRG-20800.01
P61001-P61003	YPRG-6100.01-.03	P21500	YPRG-21500
P63001, P63002	YPRG-6300.01, .02	P222001, P222002	YPRG-22200.01, .02
P9550EC	YPRG-9550-ELIG-CHECK	P228001	YPRG-22800.01
P9600	YPRG-9600	P233001, P233002	YPRG-23300.01, .02
P9700	YPRG-9700	P234001	YPRG-23400.01
P103001	YPRG-10300.01	P23900EC	YPRG-23900-ELIG-CHECK
P104001	YPRG-10400.01	P16200	YPRG-16200
P106001	YPRG-10600.01	P169001	YPRG-16900.01
P110001	YPRG-11000.01	P175001	YPRG-17500.01
P111001	YPRG-11100.01	P181001	YPRG-18100.01
P114001	YPRG-11400.01	P182001	YPRG-18200.01
P116001	YPRG-11600.01	P30500	YPRG-30500
P117001	YPRG-11700.01	P312001, P312002	YPRG-31200.01, .02
P118001	YPRG-11800.01	P318001	YPRG-31800.01
P121001	YPRG-12100.01	P323001, P323002	YPRG-32300.01, .02
P122001	YPRG-12200.01	P324001	YPRG-32400.01
P126001	YPRG-12600.01		

This program creates several variables describing the respondent's participation in government programs for the economically disadvantaged. During the interview, respondents report amounts received and months of participation in Aid to Families with Dependent Children (AFDC); food stamps; and Women, Infants, and Children (WIC). There is also an "other assistance" question to capture information about any other government program from which respondents may have received assistance. The interview also records amounts received from unemployment compensation and worker's compensation in separate question series; this program also creates variables describing respondents' participation in unemployment or worker's compensation.

The program to create these variables first creates a month-by-month participation array for each of the six categories (AFDC, food stamps, WIC, other programs, unemployment compensation, and worker's compensation). These month-by-month variables constitute part of the event history array for program participation; see appendix 7 for more information. After all six arrays are created, the program merges data from the six categories to create the summary variables.

```

cm14=((bdate_y+14)-1980)*12+bdate_m;
cmb=((bdate_y)-1980)*12+bdate_m;
ym14=(BDATE_Y*100+BDATE_M)+1400;
iym=YINT_Y*100+YINT_M;

if p1210=1 then dliym=R0000202*100+(R0000201+1) and dlicm=(R0000202-1980)*12+(R0000201+1);
if p1210 ne 1 then dliym=ym14 and dlicm=cm14;

```

```

doicm=(YINT_Y-1980)*12+YINT_M;
aaicm=(YINT_Y-BDATE_Y)*12+YINT_M-BDATE_M;

array wc      (l) wc001-wc232;
array ui      (l) ui001-ui232;
array uiAMT   (l) uiamt001-uiamt232;
array wcAMT   (l) wcamt001-wcamt232;
array ahhm    (l) ahhm001- ahhm232;
array aamt    (l) aamt001- aamt232;
array a        (l) a001- a232;
array whhm    (l) whhm001- whhm232;
array wamt    (l) wamt001- wamt232;
array w        (l) w001-w232;
array fhdm    (l) fhdm001- fhdm232;
array famt    (l) famt001- famt232;
array f        (l) f001- f232;
array ohhm    (l) ohhm001- ohhm232;
array oamt    (l) oamt001- oamt232;
array o        (l) o001- o232;

if P1210>-4 then do;
  if P21397>-3 then P21500=P21397;
  if P36087>-3 then P30500=P36087;
  if P9867>-3 then P9900=P9867;

  /*2. begin - Rs eligible for a program in round 2*/
  do l=1 to 232;
    if cmb le L le doicm then do;
      a=0; w=0; f=0; o=0; wc=0; ui=0; end;
    end;

  /** start YEAR information */
  array ysa1 (j) P358001Y P358002Y P358003Y;      /**afdc, sdli**/
  array ysa2 (j) P167001Y ysa22  ysa23;          /**afdc, int**/
  array ysw1  (j) PS22001Y PS22002Y ysw13;        /**wic, sdli**/
  array ysw2  (j) P220001Y P220002Y ysw23;        /**wic, int**/
  array ysf1   (j) P191001Y P191002Y ysf13;       /**food stamps, sdli**/
  array ysf2   (j) P194001Y P194002Y P194003Y;   /**food stamps, int**/
  array yso1   (j) PS31001Y PS31002Y yso13;       /**other, sdli**/
  array yso2   (j) P310001Y P310002Y yso23;       /**other, int**/
  array ysu1   (j) PS40001Y ysu12 ysu13;         /**unemployment, sdli**/
  array ysu2   (j) P40001Y P40002Y P40003Y;       /**unemployment, int**/
  array ysc1   (j) PS10201Y ysc12 ysc13;         /**workers comp, sdli**/
  array ysc2   (j) P102001Y ysc22 ysc23;         /**workers comp, int**/

  /** start MONTH information */
  array msa1  (j) P358001M P358002M P358003M;   /**afdc, sdli**/
  array msa2  (j) P167001M msa22  msa23;          /**afdc, int**/
  array msw1  (j) PS22001M PS22002M msw13;        /**wic, sdli**/
  array msw2  (j) P220001M P220002M msw23;        /**wic, int**/
  array msf1   (j) P191001M P191002M msf13;       /**food stamps, sdli**/
  array msf2   (j) P194001M P194002M P194003M;   /**food stamps, int**/
  array mso1   (j) PS31001M PS31002M mso13;       /**other, sdli**/
  array mso2   (j) P310001M P310002M mso23;       /**other, int**/
  array msu1   (j) PS40001M msu12 msu13;         /**unemployment, sdli**/
  array msu2   (j) P40001M P40002M P40003M;       /**unemployment, int**/
  array msc1   (j) PS10201M msc12 msc13;         /**workers comp, sdli**/

```

```

array msc2 (j) P102001M msc22 msc23;           /**workers comp, int**/

array ymsa1 (j) yms01-yms03;
array ymsa2 (j) yms04 ymsa22 ymsa23;
array ymsw1 (j) yms05-yms07;
array ymsw2 (j) yms08 yms09 ymsw23;
array ymsf1 (j) yms10 yms11 ymsf13;
array ymsf2 (j) yms12-yms14;
array ymso1 (j) yms15 yms16 ymso13;
array ymso2 (j) yms17 yms18 ymso23;
array ymsu1 (j) ymsu11-ymsu13;
array ymsu2 (j) ymsu21-ymsu23;
array ymsc1 (j) ymsc11-ymsc13;
array ymsc2 (j) ymsc21-ymsc23;

array csma1 (j) csm01-csm03;
array csma2 (j) csm04 csma22 csma23;
array csmw1 (j) csm05-csm07;
array csmw2 (j) csm08 csm09 csmw23;
array csmf1 (j) csm10 csm11 csmf13;
array csmf2 (j) csm12-csm14;
array csmo1 (j) csm15 csm16 csmo13;
array csmo2 (j) csm17 csm18 csmo23;
array csmu1 (j) csmu11-csmu13;
array csmu2 (j) csmu21-csmu23;
array csmc1 (j) csmc11-csmc13;
array csmc2 (j) csmc21-csmc23;

/** end YEAR information */
array yea1 (j) P358401Y P358402Y yea13;          /**afdc, sdli**/
array yea2 (j) P172001Y yea22  yea23;            /**afdc, int**/
array yew1 (j) PS22601Y P226002Y yew13;          /**wic, sdli**/
array yew2 (j) P226001Y yew22  yew23;            /**wic, int**/
array yef1 (j) PS20001Y PS20002Y yef13;          /**food stamps, sdli**/
array yef2 (j) P200001Y P200002Y yef23;          /**food stamps, int**/
array yeo1 (j) PS31601Y PS31602Y yeo13;          /**other, sdli**/
array yeo2 (j) P316001Y yeo22  yeo23;            /**other, int**/
array yeu1 (j) PS47001Y yeu12  yeu13;             /**unemployment, sdli**/
array yeu2 (j) P47001Y P47002Y P47003Y;          /**unemployment, int**/
array yec1 (j) PS10901Y yec12  yec13;             /**workers comp, sdil**/
array yec2 (j) P109001Y yec22  yec23;             /**workers comp, int**/

/** end MONTH information */
array meal1 (j) P358401M P358402M mea13;          /**afdc, sdli**/
array mea2 (j) P172001M mea22  mea23;            /**afdc, int**/
array mew1 (j) PS22601M P226002M mew13;          /**wic, sdli**/
array mew2 (j) P226001M mew22  mew23;            /**wic, int**/
array mef1 (j) PS20001M PS20002M mef13;          /**food stamps, sdli**/
array mef2 (j) P200001M P200002M mef23;          /**food stamps, int**/
array meo1 (j) PS31601M PS31602M meo13;          /**other, sdli**/
array meo2 (j) P316001M meo22  meo23;            /**other, int**/
array meu1 (j) PS47001M meu13;                     /**unemployment, sdli**/
array meu2 (j) P47001M P47002M P47003M;          /**unemployment, int**/
array mec1 (j) PS10901M mec12  mec13;            /**workers comp, sdil**/
array mec2 (j) P109001M mec22  mec23;            /**workers comp, int**/

array ymeal1 (j) yme01-yme03;

```

```

array ymea2 (j) yme04 ymea22 ymea23;
array ymew1 (j) yme05-yme07;
array ymew2 (j) yme08 yme09 ymew23;
array ymef1 (j) yme10 yme11 ymef13;
array ymef2 (j) yme12-yme14;
array ymeo1 (j) yme15 yme16 ymeo13;
array ymeo2 (j) yme17 yme18 ymeo23;
array ymeu1 (j) ymeu11-ymeu13;
array ymeu2 (j) ymeu21-ymeu23;
array ymec1 (j) ymec11-ymec13;
array ymec2 (j) ymec21-ymec23;

array cema1 (j) cem01-cem03;
array cema2 (j) cem04 cema22 cema23;
array cemw1 (j) cem05-cem07;
array cemw2 (j) cem08 cem09 cemw23;
array cemf1 (j) cem10 cem11 cemf13;
array cemf2 (j) cem12-cem14;
array cemo1 (j) cem15 cem16 cemo13;
array cemo2 (j) cem17 cem18 cemo23;
array cemu1 (j) cemu11-cemu13;
array cemu2 (j) cemu21-cemu23;
array cemc1 (j) cemc11-cemc13;
array cemc2 (j) cemc21-cemc23;

/** CURRENTLY receiving information */
array cura1 (j) P359501 P359502 P359503;      /**afdc, no gap sdli*/
array cura2 (j) P169001 cura22 cura23;          /**afdc, int*/
array curw1 (j) P23700A1 P23700A2 P23700A3;    /**wic, no gap sdli*/
array curw2 (j) P222001 P222002 curw23;         /**wic, int*/
array curf1 (j) P21100A1 P21100A2 curf13;       /**food stamps, no gap sdli*/
array curf2 (j) P196001 P196002 P196003;        /**food stamps, int*/
array curo1 (j) P32700A1 P32700A2 curo13;       /**other, no gap sdli*/
array curo2 (j) P312001 P312002 curo23;         /**other, int*/
array curu1 (j) PS44001 curu12 curu13;           /**unemployment, sdli*/
array curu2 (j) P44001 P44002 P44003;           /**unemployment, int*/
array curc1 (j) PS106001 curc12 curc13;          /**workers comp, sdil*/
array curc2 (j) P106001 curc22 curc23;           /**workers comp, int*/

/** EDIT FLAGS for dates */
array eflaga1 (j) eflag01-eflag03;               /**afdc, sdli*/
array eflaga2 (j) eflag04 eflaga22 eflaga23;     /**afdc, int*/
array eflagw1 (j) eflag05-eflag07;                /**wic, sdli*/
array eflagw2 (j) eflag08 eflag09 eflagw23;       /**wic, int*/
array eflagf1 (j) eflag10 eflag11 eflagf13;       /**food stamps, sdli*/
array eflagf2 (j) eflag12-eflag14;                 /**food stamps, int*/
array eflago1 (j) eflag15 eflag16 eflago13;       /**other, sdli*/
array eflago2 (j) eflag17 eflag18 eflago23;       /**other, int*/
array eflagu1 (j) eflagu11-eflagu13;              /**unemployment, sdli*/
array eflagu2 (j) eflagu21-eflagu23;              /**unemployment, int*/
array eflagc1 (j) eflagc11-eflagc13;              /**workers comp, sdil*/
array eflagc2 (j) eflagc21-eflagc23;              /**workers comp, int*/

/** EDIT FLAGS for amounts */
array aflaga1 (j) aflag01-aflag03;                /**afdc, sdli*/
array aflaga2 (j) aflag04 aflaga22 aflaga23;      /**afdc, int*/
array aflagw1 (j) aflag05-aflag07;                 /**wic, sdli*/

```

```

array aflagw2 (j) aflag08 aflag09 aflagw23;          /**wic, int**/
array aflagf1 (j) aflag10 aflag11 aflagf13;          /**food stamps, sdli**/
array aflagf2 (j) aflag12-aflag14;                  /**food stamps, int**/
array aflago1 (j) aflag15 aflag16 aflago13;          /**other, sdli**/
array aflago2 (j) aflag17 aflag18 aflago23;          /**other, int**/
array aflagu1 (j) aflagu11-aflagu13;                /**unemployment, sdli**/
array aflagu2 (j) aflagu21-aflagu23;                /**unemployment, int**/
array aflagc1 (j) aflagc11-aflagc13;                /**workers comp, sdil**/
array aflagc2 (j) aflagc21-aflagc23;                /**workers comp, int**/

/** AMOUNT RECEIVED - actual & estimated */
array inca1 (j) P359001 P359002 P359003;          /**afdc, amount sdli**/
array einca1 (j) P359101 einca12 einca13;          /**afdc, amount int**/
array inca2 (j) P181001 inca22 inca23;            /**afdc, est amount sdli**/
array einca2 (j) P182001 einca22 einca23;          /**afdc, est amount int**/
array incw1 (j) PS233001 PS233002 PS233003;        /**wic, amount sdli**/
array eincw1 (j) PS234001 PS234002 eincw13;        /**wic, amount int**/
array incw2 (j) P233001 P233002 incw23;           /**wic, est amount sdli**/
array eincw2 (j) P234001 eincw22 eincw23;          /**wic, est amount int**/
array incf1 (j) PS207001 PS207002 incf13;          /**food stamps, amount sdli**/
array eincf1 (j) PS208001 eincf12 eincf13;          /**food stamps, amount int**/
array incf2 (j) P207001 P207002 P207003;           /**food stamps, est amount sdli**/
array eincf2 (j) P208001 eincf22 eincf23;          /**food stamps, est amount int**/
array inco1 (j) PS323001 PS323002 inco13;          /**other, amount sdli**/
array einco1 (j) PS324001 einco12 einco13;          /**other, amount int**/
array inco2 (j) P323001 P323002 inco23;            /**other, est amount sdli**/
array einco2 (j) P324001 einco22 einco23;          /**other, est amount int**/

array wincu1 (j) PS59001 incu12 incu13;
array wincu2 (j) P59001 P59002 P59003;
array wincc1 (j) PS121001 incc12 incc13;
array wincc2 (j) P121001 incc22 incc23;
array eincc2 (j) P122001 eincc22 eincc23;

array wksa1 (j) wksa11-wksa13;
array wksa2 (j) P175001 wksa22 wksa23;
array wksw1 (j) PS228001 PS228002 wksw13;
array wksw2 (j) P228001 wksw22 wksw23;
array wksf1 (j) PS202001 wksf12 wksf13;
array wksf2 (j) P202001 P202002 wksf23;
array wkso1 (j) PS318001 wkso12 wkso13;
array wkso2 (j) P318001 wkso22 wkso23;
array wksu1 (j) wksu11-wksu13;
array wksu2 (j) P52001 wksu22 P52003;
array wksc1 (j) wksc11-wksc13;
array wksc2 (j) P114001 wksc22 wksc23;

array mosa1 (j) mosa11-mosa13;
array mosa2 (j) mosa21-mosa23;
array mosw1 (j) mosw11-mosw13;
array mosw2 (j) mosw21-mosw23;
array mosf1 (j) mosf11-mosf13;
array mosf2 (j) mosf21-mosf23;
array moso1 (j) moso11-moso13;
array moso2 (j) moso21-moso23;
array mosu1 (j) mosu11-mosu13;
array mosu2 (j) mosu21-mosu23;
array mosc1 (j) mosc11-mosc13;

```

```

array mosc2 (j) mosc21-mosc23;

/** PERSON receiving amount, respondent */
array rcvra1 (j) P3592011 P3592021 P3592031;      /**afdc, sdli**/
array rcvra2 (j) P1830011 rcvra22 rcvra23;          /**afdc, int**/
array rcrvw1 (j) PS235011 PS235021 PS235031;        /**wic, sdli**/
array rcrvw2 (j) P2350011 P2350021 rcrvw23;         /**wic, int**/
array rcvrf1 (j) rcvrf11-rcvrf13;                   /**food stamps, sdli**/
array rcvrf2 (j) P2090011 P2090021 P2090021;       /**food stamps, int**/
array rcvro1 (j) PS325011 PS325021 rcvro13;        /**other, sdli**/
array rcvro2 (j) P3250011 P3250021 rcvro23;         /**other, int**/

/** PERSON receiving amount, spouse */
array rcvsal (j) P3592012 P3592022 P3592032;      /**afdc, sdli**/
array rcvsal (j) P1830012 rcvsal22 rcvsal23;        /**afdc, int**/
array rcvsw1 (j) PS235012 PS235022 PS235032;        /**wic, sdli**/
array rcvsw2 (j) P2350012 P2350022 rcvsw23;         /**wic, int**/
array rcvsf1 (j) rcvsf11-rcvsf13;                   /**food stamps, sdli**/
array rcvsf2 (j) P2090012 P2090022 P2090022;       /**food stamps, int**/
array rcvso1 (j) PS325012 PS325022 rcvso13;        /**other, sdli**/
array rcvso2 (j) P3250012 P3250022 rcvso23;         /**other, int**/

/** PERSON receiving amount, child */
array rcvca1 (j) P3592013 P3592023 P3592033;      /**afdc, sdli**/
array rcvca2 (j) P1830013 rcvca22 rcvca23;          /**afdc, int**/
array rcvcw1 (j) PS235013 PS235023 PS235033;        /**wic, sdli**/
array rcvcw2 (j) P2350013 P2350023 rcvcw23;         /**wic, int**/
array rcvcf1 (j) rcvcf11-rcvcf13;                   /**food stamps, sdli**/
array rcvcf2 (j) P2090013 P2090023 P2090023;       /**food stamps, int**/
array rcvco1 (j) PS325013 PS325023 rcvco13;        /**other, sdli**/
array rcvco2 (j) P3250013 P3250023 rcvco23;         /**other, int**/

/** PERSON receiving amount family member */
array rcvfa1 (j) P3592014 P3592024 P3592034;      /**afdc, sdli**/
array rcvfa2 (j) P1830014 rcvfa22 rcvfa23;          /**afdc, int**/
array rcvfw1 (j) PS235014 PS235024 PS235034;        /**wic, sdli**/
array rcvfw2 (j) P2350014 P2350024 rcvfw23;         /**wic, int**/
array rcvff1 (j) rcvff11-rcvff13;                   /**food stamps, sdli**/
array rcvff2 (j) P2090014 P2090024 P2090024;       /**food stamps, int**/
array rcvfo1 (j) PS325014 PS325024 rcvfo13;        /**other, sdli**/
array rcvfo2 (j) P3250014 P3250024 rcvfo23;         /**other, int**/

/** PERSON receiving amount, other person */
array rcvoa1 (j) P3592015 P3592025 P3592035;      /**afdc, sdli**/
array rcvoa2 (j) P1830015 rcvoa22 rcvoa23;          /**afdc, int**/
array rcvow1 (j) PS235015 PS235025 PS235035;        /**wic, sdli**/
array rcvow2 (j) P2350015 P2350025 rcvow23;         /**wic, int**/
array rcvof1 (j) rcvof11-rcvof13;                   /**food stamps, sdli**/
array rcvof2 (j) P2090015 P2090025 P2090025;       /**food stamps, int**/
array rcvo01 (j) PS325015 PS325025 rcvo013;        /**other, sdli**/
array rcvo02 (j) P3250015 P3250025 rcvo023;         /**other, int**/

array whoa1 (j) who01-who03;
array whoa2 (j) who04 whoa22 whoa23;
array whow1 (j) who05-who07;
array whow2 (j) who08 who09 whow23;
array whof1 (j) who10 who11 whof13;

```

```

array whof2 (j) who12-who14;
array whoo1 (j) who15 who16 whoo13;
array whoo2 (j) who17 who18 whoo23;

array dlia1 (j) P357901 dlia12 dlia13;
array dliw1 (j) P21900A1 dliw12 dliw13;
array dlif1 (j) P19063A1 dlif12 dlif13;
array dlio1 (j) P30900A1 dlio12 dlio13;

/** these arrays encompass all programs for the arrays */
array ys (k) ysa1 ysa2 ysw1 ysw2 ysf1 yso1 yso2 ysu1 ysu2 ysc1 ysc2;
array ms (k) msa1 msa2 msw1 msw2 msf1 msf2 mso1 mso2 msu1 msu2 msc1 msc2;
array ye (k) yea1 yea2 yew1 yew2 yef1 yef2 yeo1 yeo2 yeu1 yeu2 yec1 yec2;
array me (k) mea1 mea2 mew1 mew2 mef1 mef2 meo1 meo2 meu1 meu2 mec1 mec2;
array cur (k) cura1 cura2 curw1 curw2 curf1 curf2 curo1 curo2 curu1 curu2 curc1 curc2;
array eflag (k) eflaga1 eflaga2 eflagw1 eflagw2 eflagf1 eflagf2 eflago1 eflago2 eflagu1 eflagu2 eflagc1 eflagc2;
array aflag (k) aflaga1 aflaga2 aflagw1 aflagw2 aflagf1 aflagf2 aflago1 aflago2 aflagu1 aflagu2 aflagc1 aflagc2;
array yms (k) ymsa1 ymsa2 ymsw1 ymsw2 ymsf1 ymsf2 yms01 yms02 ymsu1 ymsu2 ymsc1 ymsc2;
array wks (k) wksa1 wksa2 wksw1 wksw2 wksf1 wksf2 wkso1 wkso2 wksu1 wksu2 wksc1 wksc2;
array cwks (k) P35740 P16200 P21430 P21500 P16410 P18900 P36120 P30500 P1930 P3500 P9900 P9700;
array yme (k) ymea1 ymea2 ymew1 ymew2 ymef1 ymef2 ymeo1 ymeo2 ymeu1 ymeu2 ymec1 ymec2;
array mos (k) mos1 mosa2 mosw1 mosw2 mosf1 mosf2 moso1 moso2 mosu1 mosu2 mosc1 mosc2;
array csm (k) csma1 csma2 csmw1 csmw2 csmf1 csmf2 csmo1 csmo2 csmu1 csmu2 csmc1 csmc2;
array cem (k) cema1 cema2 cemw1 cemw2 cemf1 cemf2 cemo1 cemo2 cemu1 cemu2 cemc1 cemc2;
array rcvr (k) rcvra1 rcvra2 rcvrw1 rcvrw2 rcvrf1 rcvro1 rcvro2 rcvr09-rcvr12;
array rcvs (k) rcvs1 rcvs2 rcvsw1 rcvsw2 rcvsf1 rcvsf2 rcvso1 rcvs09-rcvs12;
array rcvc (k) rcvca1 rcvca2 rcvcw1 rcvcw2 rcvcf1 rcvcf2 rcvc01 rcvc02 rcvc09-rcvc12;
array rcvf (k) rcvfa1 rcvfa2 rcvfw1 rcvfw2 rcvff1 rcvff2 rcvfo1 rcvfo2 rcvf09-rcvf12;
array rcvo (k) rcvoa1 rcvoa2 rcvow1 rcvow2 rcvo1 rcvo2 rcvo01 rcvo02 rcvo09-rcvo12;
array who (k) whoa1 whoa2 whow1 whow2 whof1 whof2 whoo1 whoo2 who09-who12;
array dli (k) dlia1 dlia2 dliw1 dliw2 dlif1 dlif2 dlio1 dlio2 dli09-dli12;
array inc (k) inca1 inca2 incw1 incw2 incf1 incf2 inco1 inco2 incu1 incu2 inccl inc2;

array agefla1 (j) agefla11 agefla12 agefla13;
array agefla2 (j) agefla21 agefla22 agefla23;
array ageflw1 (j) ageflw11 ageflw12 ageflw13;
array ageflw2 (j) ageflw21 ageflw22 ageflw23;
array ageflf1 (j) ageflf11 ageflf12 ageflf13;
array ageflf2 (j) ageflf21 ageflf22 ageflf23;
array ageflo1 (j) ageflo11 ageflo12 ageflo13;
array ageflo2 (j) ageflo21 ageflo22 ageflo23;
array ageflu1 (j) ageflu11 ageflu12 ageflu13;
array ageflu2 (j) ageflu21 ageflu22 ageflu23;
array ageflc1 (j) ageflc11 ageflc12 ageflc13;
array ageflc2 (j) ageflc21 ageflc22 ageflc23;

array agefl (k) agefla1 agefla2 ageflw1 ageflw2 ageflf1 ageflf2 ageflo1 ageflo2 ageflu1 ageflu2 ageflc1 ageflc2;

NU80=0;      NU81=0;      NU82=0;      NU83=0;
NU84=0;      NU85=0;      NU86=0;      NU87=0;
NU88=0;      NU89=0;      NU90=0;      NU91=0;
NU92=0;      NU93=0;      NU94=0;      NU95=0;
NU96=0;      NU97=0;      NU98=0;      NU99=0;

```

***** This portion of the SAS program defines the start and end dates. If the respondent reports still receiving, the interview date is used as the temporary end date for the last loop reported. In the next survey round, the respondent will be asked if he or she is still receiving and, if not, a permanent end date equivalent to the interview date of the

Appendix 5: Income and Assets Variable Creation

previous round will be assigned. Users will be able to tell which method was used by looking at the following participation flag variable created during the program. The categories are the following:

1=respondent reported participation dates 2=start month imputed
3=start month and year imputed 4=stop month imputed
5=stop month and year imputed 6=start and stop dates imputed
7=error in data due to round 2 programming error *****/

```

do k=1 to 12;
do j=1 to 3;
  if ys>0 and ms>0 then yms=(ys*100)+ms;
  if dli=1 then yms=dliy;
  if cur=1 then yme=iym;
  if ye>0 and me>0 then yme=(ye*100)+me;
  if yme>iym then yme=iym;
  if yms>0 and yme ge dliy then eflag=1;
  if wks ge 0 then mos=round (wks/4.3,1);
if mos=0 then mos=1;

gef1=0;
if ys>0 and ys<1990 then agefl=1;

****1. if start year is known and month is unknown ****/
** 1.a' If weeks are known and currently receiving, then count backwards by the number of weeks from the interview date. If the number of weeks falls short of the start year, the start month is December of that year. 0If the number of weeks is past the start year, then the start month is January of that year.****/
if ys>0 and -3 le ms le -1 then do;
  if (wks>0 and cwks=1) then do;
    yme=iym;
    if mos le YINT_M then yms=iym-mos;
    if YINT_M le mos le (YINT_M+12) then yms=iym-100+(mos-12);
    if (YINT_M+12) le mos le (YINT_M+24) then yms=iym-200+(mos-24);
    if (YINT_M+24) le mos le (YINT_M+36) then yms=iym-300+(mos-36);
    if (YINT_M+36) le mos le (YINT_M+48) then yms=iym-400+(mos-48);
    if (YINT_M+48) le mos le (YINT_M+60) then yms=iym-500+(mos-60);
    if (YINT_M+60) le mos le (YINT_M+72) then yms=iym-600+(mos-72);
    if (YINT_M+72) le mos le (YINT_M+84) then yms=iym-700+(mos-84);
    if (YINT_M+84) le mos le (YINT_M+96) then yms=iym-800+(mos-96);
    if (YINT_M+96) le mos le (YINT_M+108) then yms=iym-900+(mos-108);
    if (YINT_M+108) le mos le (YINT_M+120) then yms=iym-1000+(mos-120);
    if yms<((ys*100)+01) then yms=((ys*100)+01);
    if yms<dliy then yms=dliy;
    if yms>((ys*100)+12) then yms=((ys*100)+12);
  end;
**weeks missing and continuously receiving - set end date to current interview date and start date to December of start year**
  else if (-3 le wks le 0 and cwks=1) then do;
    yme=iym;
    yms=((ys*100)+12);
    if yms>iym then yms=iym;
  end;
**weeks missing and not continuously receiving - set end date to December and start date to January of start year**
  else if (-3 le wks le 0 and cwks=0) then do;
    yms=((ys*100)+01);
    yme=((ys*100)+12);

```

```

if yme>iym then yme=iym;
end;

/** 1.a" if weeks are known and not currently receiving, then count forward by the number of weeks from January
of the start year. If the count exceeds the interview date then stop counting at the interview date.****/
else if (wks>0 and cwks=0) then do;
    yms=((ys*100)+01);
    if yms<dliym then yms=dliym;
    if 0 le mos le 12 then yme=yms+mos;
    if wks>0 and cwks=0 and 13 le mos le 24 then yme=((ys+1)*100)+mos-12;
    if wks>0 and cwks=0 and 25 le mos le 36 then yme=((ys+2)*100)+mos-24;
    if wks>0 and cwks=0 and 37 le mos le 48 then yme=((ys+3)*100)+mos-36;
    if wks>0 and cwks=0 and 49 le mos le 60 then yme=((ys+4)*100)+mos-48;
    if wks>0 and cwks=0 and 61 le mos le 72 then yme=((ys+5)*100)+mos-60;
    if wks>0 and cwks=0 and 73 le mos le 84 then yme=((ys+6)*100)+mos-72;
    if wks>0 and cwks=0 and 85 le mos le 96 then yme=((ys+7)*100)+mos-84;
    if wks>0 and cwks=0 and 97 le mos le 108 then yme=((ys+8)*100)+mos-96;
    if wks>0 and cwks=0 and 109 le mos le 120 then yme=((ys+9)*100)+mos-108;
    if yme>iym then yme=iym;
end;

if yme>0 and cwks=0 then yms=((ys*100)+01);
if yms<dliym then yms=dliym;
eflag=2;
end;

*****2. if start year is unknown but weeks are known then count back from interview date if currently receiving. If
not currently receiving, then count back from interview date to find the most recent year the respondent could have
begun receiving and receive for that number of months; then count forward the number of months from January of
that year ****/
if -3 le ys le -1 then do;
    if (wks>0 and cwks=1) then do;
        if 0 le mos le YINT_M then yms=iym-mos;
        if YINT_M le mos le (YINT_M+12) then yms=iym-100-(mos-12);
        if YINT_M+12 le mos le (YINT_M+24) then yms=iym-200-(mos-24);
        if (YINT_M+24) le mos le (YINT_M+36) then yms=iym-300-(mos-36);
        if (YINT_M+36) le mos le (YINT_M+48) then yms=iym-400-(mos-48);
        if (YINT_M+48) le mos le (YINT_M+60) then yms=iym-500-(mos-60);
        if (YINT_M+60) le mos le (YINT_M+72) then yms=iym-600-(mos-72);
        if (YINT_M+72) le mos le (YINT_M+84) then yms=iym-700-(mos-84);
        if (YINT_M+84) le mos le (YINT_M+96) then yms=iym-800-(mos-96);
        if (YINT_M+96) le mos le (YINT_M+108) then yms=iym-900-(mos-108);
        if (YINT_M+108) le mos le (YINT_M+120) then yms=iym-1000-(mos-120);
        yme=iym;
    end;

    if (-3 le wks le 0 and cwks=1) then do;
        yms=dliym;
        yme=iym;
    end;

    if (wks>0 and cwks=0) then do;
        if 0 le mos le YINT_M then do;
            yms=(YINT_Y*100)+01;
            yme=(YINT_Y*100)+01+mos;
        end;
    end;

```

```

if (YINT_M+01) le mos le (YINT_M+12) then do;
  yms=((YINT_Y-1)*100)+01;
  if (YINT_M+01) le mos le 12 then yme=((YINT_Y-1)*100)+01+mos;
  if 13 le mos le (YINT_M+12) then yme=(YINT_Y*100)+01+mos;
end;
if (YINT_M+13) le mos le (YINT_M+24) then do;
  yms=((YINT_Y-2)*100)+01;
  if (YINT_M+13) le mos le 24 then yme=((YINT_Y-1)*100)+01+mos;
  if 25 le mos le (YINT_M+24) then yme=(YINT_Y*100)+01+mos;
end;
if yme>iym then yme=iym;
end;
eflag=3;
end;

if yme>0 and yms=. then do;
  yms=dliym;
  if j=1 then eflag=3;
  else if j>1 then eflag=7;
end;

```

*****3. If stop year is known and weeks are known and not currently receiving, but stop month is not known, then count forward from start year. If the number of months falls short of the stop year, then use January of the end year as the stop date; if the number of months exceeds the stop year, then end the array in the December of the stop year. If the stop year is equal to the interview year and the stop month exceeds the interview month, then stop at the interview date. If currently receiving, use interview date as the stop date. *****

```

if yms>0 and ye>0 and -2 le me le -1 then do;
  if wks>0 then do;
    if 01 le ((yms- round (yms,100))+mos) le 12 then yme=yms+mos;
    if 12 lt ((yms- round (yms,100))+mos) le 24 then yme=yms+100+mos-12;
    if 24 lt ((yms- round (yms,100))+mos) le 36 then yme=yms+200+mos-24;
    if yme>((ye*100)+12) then yme=((ye*100)+12);
    if yme<((ye*100)+01) then yme=((ye*100)+01);
    if yme>iym then yme=iym;
  end;
  if -3 le wks le 0 then yme=((ye*100)+12);
  if yme>iym then yme=iym;
  eflag=4;
end;

```

*** 4. if stop year is unknown, and weeks are known***

```

if -3 le ye le -1 then do;
  if yms>0 and wks>0 then do;
    if 01 le ((yms- round (yms,100))+mos) le 12 then yme=yms+mos;
    if 12 lt ((yms- round (yms,100))+mos) le 24 then yme=yms+100+mos-12;
    if 24 lt ((yms- round (yms,100))+mos) le 36 then yme=yms+200+mos-24;
    if yme>iym then yme=iym;
  end;

```

*** 5. if stop year is unknown, and weeks are unknown***

```

if yms>0 and -3 le wks le 0 then yme=(round (yms,100))+12;
eflag=5;
end;

```

*** 6. if the start and stop years are unknown and the weeks are unknown, use current and last interview date***

```

if -3 le ys le -1 and cwks=0 and -3 le wks le 0 then do;
  yme=iym;

```

```

yms=dliym;
if j=1 then eflag=6;
else if j>1 then eflag=7;
end;

```

***** This portion of the program creates a variable that determines who receives afdc in the household. It collapses the answers to 8 categories. The coding is the following -

1=respondent only	5=respondent and child
2=spouse/partner only	6=spouse/partner and child
3=child only	7=respondent and spouse/partner and child
4=respondent and spouse/partner	8=other

The first 7 categories may include an 'other' person as captured by response categories 4 and 5 in the original question. If only another person is listed as receiving, then the 8th answer category is used in the created variable. **/

```

if rcvr=1 and rcvs=0 and rcvc=0 then who=01;
if rcvr=0 and rcvs=1 and rcvc=0 then who=02;
if rcvr=0 and rcvs=0 and rcvc=1 then who=03;
if rcvr=1 and rcvs=1 and rcvc=0 then who=04;
if rcvr=1 and rcvs=0 and rcvc=1 then who=05;
if rcvr=0 and rcvs=1 and rcvc=1 then who=06;
if rcvr=1 and rcvs=1 and rcvc=1 then who=07;
if -3 le rcvr lt 0 then who=rcvr;
if rcvr=0 and rcvs=0 and rcvc=0 and (rcvf=1 or rcvo=1) then who=08;

```

***** This portion of the program uses the category reported by the respondent to create an estimated amount. The estimated amount is the midpoint rounded down. Note that the 12th category lists \$1251 as the amount. This amount was chosen since the category is unbounded - the number represents one dollar above the lower bound.****/

```

if -3 le incf1 le -1 and 1 le eincf1 le 10 then incf1=(eincf1*100)-50;
if -3 le incf1 le -1 and eincf1=11 then incf1=1125;
if -3 le incf1 le -1 and eincf1=12 then incf1=1251;
if -3 le incf2 le -1 and 1 le eincf2 le 10 then incf2=(eincf2*100)-50;
if -3 le incf2 le -1 and eincf2=11 then incf2=1125;
if -3 le incf2 le -1 and eincf2=12 then incf2=1251;
if -3 le inca1 le -1 and 1 le einca1 le 10 then inca1=(einca1*100)-50;
if -3 le inca1 le -1 and einca1=11 then inca1=1125;
if -3 le inca1 le -1 and einca1=12 then inca1=1251;
if -3 le inca2 le -1 and 1 le einca2 le 10 then inca2=(einca2*100)-50;
if -3 le inca2 le -1 and einca2=11 then inca2=1125;
if -3 le inca2 le -1 and einca2=12 then inca2=1251;
if -3 le inco1 le -1 and 1 le einco1 le 10 then inco1=(einco1*100)-50;
if -3 le inco1 le -1 and einco1=11 then inco1=1125;
if -3 le inco1 le -1 and einco1=12 then inco1=1251;
if -3 le inco2 le -1 and 1 le einco2 le 10 then inco2=(einco2*100)-50;
if -3 le inco2 le -1 and einco2=11 then inco2=1125;
if -3 le inco2 le -1 and einco2=12 then inco2=1251;
if -3 le incw1 le -1 and 1 le eincw1 le 5 then incw1=(eincw1*20)-10;
if -3 le incw1 le -1 and eincw1=6 then incw1=101;
if -3 le incw2 le -1 and 1 le eincw2 le 5 then incw2=(eincw2*20)-10;
if -3 le incw2 le -1 and eincw2=6 then incw2=101;
if wincu1>0 then incu1= round ((wincu1*52)/12,1);
if wincu1<0 then incu1=wincu1;
if wincu2>0 then incu2= round ((wincu2*52)/12,1);
if wincu2<0 then incu2=wincu2;
if wincc1>0 then inccl= round ((wincc1*52)/12,1);
if wincc1<0 then inccl=wincc1;
if wincc2>0 then inccl= round ((wincc2*52)/12,1);
if wincc2<0 then inccl=wincc2;

```

```

if inc>-4 then aflag=0;

csm=(round(yms,100)-198000)*.12+(yms-round(yms,100));
cem=(round(yme,100)-198000)*.12+(yme-round(yme,100));

*****to compute the spells of ui*****
if 9 le K le 10 then DO;
  if 1 le CSM le 12 then NU80=NU80+1;
  if 13 le CSM le 24 then NU81=NU81+1;
  if 25 le CSM le 36 then NU82=NU82+1;
  if 37 le CSM le 48 then NU83=NU83+1;
  if 49 le CSM le 60 then NU84=NU84+1;
  if 61 le CSM le 72 then NU85=NU85+1;
  if 73 le CSM le 84 then NU86=NU86+1;
  if 85 le CSM le 96 then NU87=NU87+1;
  if 97 le CSM le 108 then NU88=NU88+1;
  if 109 le CSM le 120 then NU89=NU89+1;
  if 121 le CSM le 132 then NU90=NU90+1;
  if 133 le CSM le 144 then NU91=NU91+1;
  if 145 le CSM le 156 then NU92=NU92+1;
  if 157 le CSM le 168 then NU93=NU93+1;
  if 169 le CSM le 180 then NU94=NU94+1;
  if 181 le CSM le 192 then NU95=NU95+1;
  if 193 le CSM le 204 then NU96=NU96+1;
  if 205 le CSM le 216 then NU97=NU97+1;
  if 217 le CSM le 228 then NU98=NU98+1;
  if 229 le CSM le 232 then NU99=NU99+1;
end;

C=0;
do L=1 to 232;
  C=C+1;
  if 0 le L le doicm then do;
    if csm LE C LE cem then do;
      if 1 le k le 2 then do;
        a=eflag; ahhm=who; aamt=inc;
        if aamt>1229 then do;
          amtodd=1;
        end;
      end;
      if 3 le k le 4 then do;
        w=eflag; whhm=who; wamt=inc;
      end;
      if 5 le k le 6 then do;
        f=eflag; fhdm=who; famt=inc;
        if famt>1000 then do;
          amtodd=2;
        end;
      end;
      if 7 le k le 8 then do;
        o=eflag; ohhm=who; oamt=inc;
      end;
      if 9 le k le 10 then do;
        ui=eflag; uiamt=inc;
      end;
      if 11 le k le 12 then do;
        wc=eflag; wcamt=inc;
      end;
    end;
  end;
end;

```



```

wa4=0; wn4=0; wa5=0; wn5=0; wa6=0; wn6=0;
ua7=0; un7=0; ua8=0; un8=0; ua9=0; un9=0;
wa7=0; wn7=0; wa8=0; wn8=0; wa9=0; wn9=0;
incprg=0; prgamt=0; nnp=0; nap=0; out=0;

/*total months - part 1*/
do L=1 to 232;
  if 1 le L le 232 then do;
    if TTLM ge 0 and A>0 or W>0 or F>0 or O>0
      then do;
        TTLM=TTLM+1;
      end;
    if a>1 then TTLM=-3;
    if w>1 then TTLM=-3;
    if f>1 then TTLM=-3;
    if o>1 then TTLM=-3;
    if TTLU ge 0 and UI>0 then do;
      TTLU=TTLU+1;
    end;
    if UI>1 then TTLU=-3;
    if TTLW ge 0 and WC>0 then do;
      TTLW=TTLW+1;
    end;
    if WC>1 then TTLW=-3;
  end;

  if 1 le L le 12 then do;
    if M80 ge 0 and a>0 or w>0 or f>0 or o>0 then do;
      M80=M80+1;
    end;
    if a>1 then M80=-3;
    if w>1 then M80=-3;
    if f>1 then M80=-3;
    if o>1 then M80=-3;
    array allamt80 aamt wamt famt oamt;
    do over allamt80;
      if -3 le allamt80 le -1 then do;
        nn80=nn80+1;
        amt80=allamt80;
      end;
    if allamt80 ge 0 then do;
      na80=na80+1;
      if na80=0 and nn80=0 then amt80=allamt80;
      if na80>1 and nn80=0 then
        amt80=amt80+allamt80;
      if M80=-3 then amt80=-3;
    end;
  end;
  if UM80 ge 0 and UI>0 then do;
    UM80=UM80+1;
  end;
  if UI>1 then UM80=-3;
  if -3 le uiamt le -1 then do;
    un80=un80+1;
    uamt80=uiamt;
  end;
  if uiamt ge 0 then do;
    ua80=ua80+1;
    if ua80=1 and un80=0 then UAMT80=UIAMT;
    if ua80>1 and un80=0 then
      uamt80=uamt80+uiamt;
    if UM80=-3 then uamt80=-3;
  end;
  if WM80 ge 0 and WC>0 then do;
    WM80=WM80+1;
  end;
  if WC>1 then WM80=-3;
  if -3 le wcamt le -1 then do;
    wn80=wn80+1;
    camt80=wcamt;
  end;
  if wcamt ge 0 then do;
    wa80=wa80+1;
    if wa80=1 and wn80=0 then
      cAMT80=WCAMT;
    if wa80>1 and wn80=0 then
      camt80=camt80+wcamt;
    if WM80=-3 then camt80=-3;
  end;
  if 13 le L le 24 then do;
    if M81 ge 0 and a>0 or w>0 or f>0 or o>0 then do;
      M81=M81+1;
    end;
    if a>1 then M81=-3;
    if w>1 then M81=-3;
    if f>1 then M81=-3;
    if o>1 then M81=-3;
    array allamt81 aamt wamt famt oamt;
    do over allamt81;
      if -3 le allamt81 le -1 then do;
        nn81=nn81+1;
        amt81=allamt81;
      end;
    if allamt81 ge 0 then do;
      na81=na81+1;
      if na81=0 and nn81=0 then amt81=allamt81;
      if na81>1 and nn81=0 then
        amt81=amt81+allamt81;
      if M81=-3 then amt81=-3;
    end;
  end;
  if UM81 ge 0 and UI>0 then do;
    UM81=UM81+1;
  end;
  if UI>1 then UM81=-3;
  if -3 le uiamt le -1 then do;
    un81=un81+1;
  end;

```

<pre> uamt81=uiamt; end; if uiamt ge 0 then do; ua81=ua81+1; if ua81=1 and un81=0 then UAMT81=UIAMT; if ua81>1 and un81=0 then uamt81=uamt81+uiamt; if UM81=-3 then uamt81=-3; end; if WM81 ge 0 and WC>0 then do; WM81=WM81+1; end; if WC>1 then WM81=-3; if -3 le wcamt le -1 then do; wn81=wn81+1; camt81=wcamt; end; if wcamt ge 0 then do; wa81=wa81+1; if wa81=1 and wn81=0 then cAMT81=WCAMT; if wa81>1 and wn81=0 then camt81=camt81+wcamt; if WM81=-3 then camt81=-3; end; end; if 25 le L le 36 then do; if M82 ge 0 and a>0 or w>0 or f>0 or o>0 then do; M82=M82+1; end; if a>1 then M82=-3; if w>1 then M82=-3; if f>1 then M82=-3; if o>1 then M82=-3; array allamt82 aamt wamt famt oamt; do over allamt82; if -3 le allamt82 le -1 then do; nn82=nn82+1; amt82=allamt82; end; if allamt82 ge 0 then do; na82=na82+1; if na82=0 and nn82=0 then amt82=allamt82; if na82>1 and nn82=0 then amt82=amt82+allamt82; if M82=-3 then amt82=-3; end; end; if UM82 ge 0 and UI>0 then do; UM82=UM82+1; end; if UI>1 then UM82=-3; if -3 le uiamt le -1 then do; un82=un82+1; uamt82=uiamt; end; if uiamt ge 0 then do; </pre>	<pre> ua82=ua82+1; if ua82=1 and un82=0 then UAMT82=UIAMT; if ua82>1 and un82=0 then uamt82=uamt82+uiamt; if UM82=-3 then uamt82=-3; end; if WM82 ge 0 and WC>0 then do; WM82=WM82+1; end; if WC>1 then WM82=-3; if -3 le wcamt le -1 then do; wn82=wn82+1; camt82=wcamt; end; if wcamt ge 0 then do; wa82=wa82+1; if wa82=1 and wn82=0 then cAMT82=WCAMT; if wa82>1 and wn82=0 then camt82=camt82+wcamt; if WM82=-3 then camt82=-3; end; end; if 37 le L le 48 then do; if M83 ge 0 and a>0 or w>0 or f>0 or o>0 then do; M83=M83+1; end; if a>1 then M83=-3; if w>1 then M83=-3; if f>1 then M83=-3; if o>1 then M83=-3; array allamt83 aamt wamt famt oamt; do over allamt83; if -3 le allamt83 le -1 then do; nn83=nn83+1; amt83=allamt83; end; if allamt83 ge 0 then do; na83=na83+1; if na83=0 and nn83=0 then amt83=allamt83; if na83>1 and nn83=0 then amt83=amt83+allamt83; if M83=-3 then amt83=-3; end; end; if UM83 ge 0 and UI>0 then do; UM83=UM83+1; end; if UI>1 then UM83=-3; if -3 le uiamt le -1 then do; un83=un83+1; uamt83=uiamt; end; if uiamt ge 0 then do; ua83=ua83+1; if ua83=1 and un83=0 then UAMT83=UIAMT; </pre>
--	---

```

if ua83>1 and un83=0 then
    uamt83=uamt83+uiamt;
    if UM83=-3 then uamt83=-3;
end;
if WM83 ge 0 and WC>0 then do;
    WM83=WM83+1;
end;
if WC>1 then WM83=-3;
if -3 le wcamt le -1 then do;
    wn83=wn83+1;
    camt83=wcamt;
end;
if wcamt ge 0 then do;
    wa83=wa83+1;
    if wa83=1 and wn83=0 then cAMT83=WCAMT;
    if wa83>1 and wn83=0 then
        camt83=camt83+wcamt;
    if WM83=-3 then camt83=-3;
end;
end;

if 49 le L le 60 then do;
if M84 ge 0 and a>0 or w>0 or f>0 or o>0 then do;
    M84=M84+1;
end;
if a>1 then M84=-3;
if w>1 then M84=-3;
if f>1 then M84=-3;
if o>1 then M84=-3;
array allamt84 aamt wamt famt oamt;
do over allamt84;
    if -3 le allamt84 le -1 then do;
        nn84=nn84+1;
        amt84=allamt84;
    end;
    if allamt84 ge 0 then do;
        na84=na84+1;
        if na84=0 and nn84=0 then amt84=allamt84;
        if na84>1 and nn84=0 then
            amt84=amt84+allamt84;
        if M84=-3 then amt84=-3;
    end;
end;
if UM84 ge 0 and UI>0 then do;
    UM84=UM84+1;
end;
if UI>1 then UM84=-3;
if -3 le uiamt le -1 then do;
    un84=un84+1;
    uamt84=uiamt;
end;
if uiamt ge 0 then do;
    ua84=ua84+1;
    if ua84=1 and un84=0 then UAMT84=UIAMT;
    if ua84>1 and un84=0 then
        uamt84=uamt84+uiamt;
    if UM84=-3 then uamt84=-3;
end;

```

```

end;
if WM84 ge 0 and WC>0 then do;
    WM84=WM84+1;
end;
if WC>1 then WM84=-3;
if -3 le wcamt le -1 then do;
    wn84=wn84+1;
    camt84=wcamt;
end;
if wcamt ge 0 then do;
    wa84=wa84+1;
    if wa84=1 and wn84=0 then cAMT84=WCAMT;
    if wa84>1 and wn84=0 then
        camt84=camt84+wcamt;
    if WM84=-3 then camt84=-3;
end;
end;

if 61 le L le 72 then do;
if M85 ge 0 and a>0 or w>0 or f>0 or o>0 then do;
    M85=M85+1;
end;
if a>1 then M85=-3;
if w>1 then M85=-3;
if f>1 then M85=-3;
if o>1 then M85=-3;
array allamt85 aamt wamt famt oamt;
do over allamt85;
    if -3 le allamt85 le -1 then do;
        nn85=nn85+1;
        amt85=allamt85;
    end;
    if allamt85 ge 0 then do;
        na85=na85+1;
        if na85=0 and nn85=0 then amt85=allamt85;
        if na85>1 and nn85=0 then
            amt85=amt85+allamt85;
        if M85=-3 then amt85=-3;
    end;
end;
if UM85 ge 0 and UI>0 then do;
    UM85=UM85+1;
end;
if UI>1 then UM85=-3;
if -3 le uiamt le -1 then do;
    un85=un85+1;
    uamt85=uiamt;
end;
if uiamt ge 0 then do;
    ua85=ua85+1;
    if ua85=1 and un85=0 then UAMT85=UIAMT;
    if ua85>1 and un85=0 then
        uamt85=uamt85+uiamt;
    if UM85=-3 then uamt85=-3;
end;
if WM85 ge 0 and WC>0 then do;
    WM85=WM85+1;

```

<pre> end; if WC>1 then WM85=-3; if -3 le wcamt le -1 then do; wn85=wn85+1; camt85=wcamt; end; if wcamt ge 0 then do; wa85=wa85+1; if wa85=1 and wn85=0 then cAMT85=WCAMT; if wa85>1 and wn85=0 then camt85=camt85+wcamt; if WM85=-3 then camt85=-3; end; end; if 73 le L le 84 then do; if M86 ge 0 and a>0 or w>0 or f>0 or o>0 then do; M86=M86+1; end; if a>1 then M86=-3; if w>1 then M86=-3; if f>1 then M86=-3; if o>1 then M86=-3; array allamt86 aamt wamt famt oamt; do over allamt86; if -3 le allamt86 le -1 then do; nn86=nn86+1; amt86=allamt86; end; if allamt86 ge 0 then do; na86=na86+1; if na86=0 and nn86=0 then amt86=allamt86; if na86>1 and nn86=0 then amt86=amt86+allamt86; if M86=-3 then amt86=-3; end; end; if UM86 ge 0 and UI>0 then do; UM86=UM86+1; end; if UI>1 then UM86=-3; if -3 le uiamt le -1 then do; un86=un86+1; uamt86=uiamt; end; if uiamt ge 0 then do; ua86=ua86+1; if ua86=1 and un86=0 then UAMT86=UIAMT; if ua86>1 and un86=0 then uamt86=uamt86+uiamt; if UM86=-3 then uamt86=-3; end; if WM86 ge 0 and WC>0 then do; WM86=WM86+1; end; if WC>1 then WM86=-3; if -3 le wcamt le -1 then do; </pre>	<pre> wn86=wn86+1; camt86=wcamt; end; if wcamt ge 0 then do; wa86=wa86+1; if wa86=1 and wn86=0 then cAMT86=WCAMT; if wa86>1 and wn86=0 then camt86=camt86+wcamt; if WM86=-3 then camt86=-3; end; end; if 85 le L le 96 then do; if M87 ge 0 and a>0 or w>0 or f>0 or o>0 then do; M87=M87+1; end; if a>1 then M87=-3; if w>1 then M87=-3; if f>1 then M87=-3; if o>1 then M87=-3; array allamt87 aamt wamt famt oamt; do over allamt87; if -3 le allamt87 le -1 then do; nn87=nn87+1; amt87=allamt87; end; if allamt87 ge 0 then do; na87=na87+1; if na87=0 and nn87=0 then amt87=allamt87; if na87>1 and nn87=0 then amt87=amt87+allamt87; if M87=-3 then amt87=-3; end; end; if UM87 ge 0 and UI>0 then do; UM87=UM87+1; end; if UI>1 then UM87=-3; if -3 le uiamt le -1 then do; un87=un87+1; uamt87=uiamt; end; if uiamt ge 0 then do; ua87=ua87+1; if ua87=1 and un87=0 then UAMT87=UIAMT; if ua87>1 and un87=0 then uamt87=uamt87+uiamt; if UM87=-3 then uamt87=-3; end; if WM87 ge 0 and WC>0 then do; WM87=WM87+1; end; if WC>1 then WM87=-3; if -3 le wcamt le -1 then do; wn87=wn87+1; camt87=wcamt; end; </pre>
--	--

<pre> if wcamt ge 0 then do; wa87=wa87+1; if wa87=1 and wn87=0 then cAMT87=WCAMT; if wa87>1 and wn87=0 then camt87=camt87+wcamt; if WM87=-3 then camt87=-3; end; end; if 97 le L le 108 then do; if M88 ge 0 and a>0 or w>0 or f>0 or o>0 then do; M88=M88+1; end; if a>1 then M88=-3; if w>1 then M88=-3; if f>1 then M88=-3; if o>1 then M88=-3; array allamt88 aamt wamt famt oamt; do over allamt88; if -3 le allamt88 le -1 then do; nn88=nn88+1; amt88=allamt88; end; if allamt88 ge 0 then do; na88=na88+1; if na88=0 and nn88=0 then amt88=allamt88; if na88>1 and nn88=0 then amt88=amt88+allamt88; if M88=-3 then amt88=-3; end; end; if UM88 ge 0 and UI>0 then do; UM88=UM88+1; end; if UI>1 then UM88=-3; if -3 le uiamt le -1 then do; un88=un88+1; uamt88=uiamt; end; if uiamt ge 0 then do; ua88=ua88+1; if ua88=1 and un88=0 then UAMT88=UIAMT; if ua88>1 and un88=0 then uamt88=uamt88+uiamt; if UM88=-3 then uamt88=-3; end; if WM88 ge 0 and WC>0 then do; WM88=WM88+1; end; if WC>1 then WM88=-3; if -3 le wcamt le -1 then do; wn88=wn88+1; camt88=wcamt; end; if wcamt ge 0 then do; wa88=wa88+1; if wa88=1 and wn88=0 then cAMT88=WCAMT; </pre>	<pre> if wa88>1 and wn88=0 then camt88=camt88+wcamt; if WM88=-3 then camt88=-3; end; end; if 109 le L le 120 then do; if M89 ge 0 and a>0 or w>0 or f>0 or o>0 then do; M89=M89+1; end; if a>1 then M89=-3; if w>1 then M89=-3; if f>1 then M89=-3; if o>1 then M89=-3; array allamt89 aamt wamt famt oamt; do over allamt89; if -3 le allamt89 le -1 then do; nn89=nn89+1; amt89=allamt89; end; if allamt89 ge 0 then do; na89=na89+1; if na89=0 and nn89=0 then amt89=allamt89; if na89>1 and nn89=0 then amt89=amt89+allamt89; if M89=-3 then amt89=-3; end; end; if UM89 ge 0 and UI>0 then do; UM89=UM89+1; end; if UI>1 then UM89=-3; if -3 le uiamt le -1 then do; un89=un89+1; uamt89=uiamt; end; if uiamt ge 0 then do; ua89=ua89+1; if ua89=1 and un89=0 then UAMT89=UIAMT; if ua89>1 and un89=0 then uamt89=uamt89+uiamt; if UM89=-3 then uamt89=-3; end; if WM89 ge 0 and WC>0 then do; WM89=WM89+1; end; if WC>1 then WM89=-3; if -3 le wcamt le -1 then do; wn89=wn89+1; camt89=wcamt; end; if wcamt ge 0 then do; wa89=wa89+1; if wa89=1 and wn89=0 then cAMT89=WCAMT; if wa89>1 and wn89=0 then camt89=camt89+wcamt; if WM89=-3 then camt89=-3; end; </pre>
---	--

<pre> end; end; if 121 le L le 132 then do; if M90 ge 0 and a>0 or w>0 or f>0 or o>0 then do; M90=M90+1; end; if a>1 then M90=-3; if w>1 then M90=-3; if f>1 then M90=-3; if o>1 then M90=-3; array allamt0 aamt wamt famt oamt; do over allamt0; if -3 le allamt0 le -1 then do; nn0=nn0+1; amt90=allamt0; end; if allamt0 ge 0 then do; na0=na0+1; if na0=0 and nn0=0 then amt90=allamt0; if na0>1 and nn0=0 then amt90=amt90+allamt0; if M90=-3 then amt90=-3; end; end; if UM90 ge 0 and UI>0 then do; UM90=UM90+1; end; if UI>1 then UM90=-3; if -3 le uiamt le -1 then do; un0=un0+1; uamt90=uiamt; end; if uiamt ge 0 then do; ua0=ua0+1; if ua0=1 and un0=0 then UAMT90=UIAMT; if ua0>1 and un0=0 then uamt90=uamt90+uiamt; if UM90=-3 then uamt90=-3; end; if WM90 ge 0 and WC>0 then do; WM90=WM90+1; end; if WC>1 then WM90=-3; if -3 le wcamt le -1 then do; wn0=wn0+1; camt90=wcamt; end; if wcamt ge 0 then do; wa0=wa0+1; if wa0=1 and wn0=0 then cAMT90=WCAMT; if wa0>1 and wn0=0 then camt90=camt90+wcamt; if WM90=-3 then camt90=-3; end; end; if 133 le L le 144 then do; if M91 ge 0 and a>0 or w>0 or f>0 or o>0 then do; </pre>	<pre> M91=M91+1; end; if a>1 then M91=-3; if w>1 then M91=-3; if f>1 then M91=-3; if o>1 then M91=-3; array allamt1 aamt wamt famt oamt; do over allamt1; if -3 le allamt1 le -1 then do; nn1=nn1+1; amt91=allamt1; end; if allamt1 ge 0 then do; na1=na1+1; if na1=1 and nn1=0 then amt91=allamt1; if na1>1 and nn1=0 then amt91=amt91+allamt1; if M91=-3 then amt91=-3; end; if UI>1 then UM91=-3; if -3 le uiamt le -1 then do; un1=un1+1; uamt91=uiamt; end; if uiamt ge 0 then do; ua1=ua1+1; if ua1=1 and un1=0 then UAMT91=UIAMT; if ua1>1 and un1=0 then uamt91=uamt91+uiamt; if UM91=-3 then uamt91=-3; end; if WM91 ge 0 and WC>0 then do; WM91=WM91+1; end; if WC>1 then WM91=-3; if -3 le wcamt le -1 then do; wn1=wn1+1; camt91=wcamt; end; if wcamt ge 0 then do; wa1=wa1+1; if wa1=1 and wn1=0 then cAMT91=WCAMT; if wa1>1 and wn1=0 then camt91=camt91+wcamt; if WM91=-3 then camt91=-3; end; end; if 145 le L le 156 then do; if M92 ge 0 and a>0 or w>0 or f>0 or o>0 then do; M92=M92+1; end; if a>1 then M92=-3; if w>1 then M92=-3; if f>1 then M92=-3; </pre>
--	---

```

if o>1 then M92=-3;
array allamt2 aamt wamt famt oamt;
do over allamt2;
  if -3 le allamt2 le -1 then do;
    nn2=nn2+1;
    amt92=allamt2;
  end;
  if allamt2 ge 0 then do;
    na2=na2+1;
    if na2=1 and nn2=0 then amt92=allamt2;
    if na2>1 and nn2=0 then amt92=amt92+allamt2;
    if M92=-3 then amt92=-3;
  end;
end;
if UM92 ge 0 and UI>0 then do;
  UM92=UM92+1;
end;
if UI>1 then UM92=-3;
if -3 le uiamt le -1 then do;
  un2=un2+1;
  uamt92=uiamt;
end;
if uiamt ge 0 then do;
  ua2=ua2+1;
  if ua2=1 and un2=0 then UAMT92=UIAMT;
  if ua2>1 and un2=0 then uamt92=uamt92+uiamt;
  if UM92=-3 then uamt92=-3;
end;
if WM92 ge 0 and WC>0 then do;
  WM92=WM92+1;
end;
if WC>1 then WM92=-3;
if -3 le wcamt le -1 then do;
  wn2=wn2+1;
  camt92=wcamt;
end;
if wcamt ge 0 then do;
  wa2=wa2+1;
  if wa2=1 and wn2=0 then cAMT92=WCAMT;
  if wa2>1 and wn2=0 then
    camt92=camt92+wcamt;
  if WM92=-3 then camt92=-3;
end;
end;

if 157 le L le 168 then do;
  if M93 ge 0 and a>0 or w>0 or f>0 or o>0 then do;
    M93=M93+1;
  end;
  if a>1 then M93=-3;
  if w>1 then M93=-3;
  if f>1 then M93=-3;
  if o>1 then M93=-3;
  array allamt3 aamt wamt famt oamt;
  do over allamt3;
    if -3 le allamt3 le -1 then do;
      nn3=nn3+1;

```

```

      amt93=allamt3;
    end;
    if allamt3 ge 0 then do;
      na3=na3+1;
      if na3=1 and nn3=0 then amt93=allamt3;
      if na3>1 and nn3=0 then amt93=amt93+allamt3;
      if M93=-3 then amt93=-3;
    end;
    end;
    if UM93 ge 0 and UI>0 then do;
      UM93=UM93+1;
    end;
    if UI>1 then UM93=-3;
    if -3 le uiamt le -1 then do;
      un3=un3+1;
      uamt93=uiamt;
    end;
    if uiamt ge 0 then do;
      ua3=ua3+1;
      if ua3=1 and un3=0 then UAMT93=UIAMT;
      if ua3>1 and un3=0 then uamt93=uamt93+uiamt;
      if UM93=-3 then uamt93=-3;
    end;
    if WM93 ge 0 and WC>0 then do;
      WM93=WM93+1;
    end;
    if WC>1 then WM93=-3;
    if -3 le wcamt le -1 then do;
      wn3=wn3+1;
      camt93=wcamt;
    end;
    if wcamt ge 0 then do;
      wa3=wa3+1;
      if wa3=1 and wn3=0 then cAMT93=WCAMT;
      if wa3>1 and wn3=0 then
        camt93=camt93+wcamt;
      if WM93=-3 then camt93=-3;
    end;
    end;

    if 169 le L le 180 then do;
      if M94 ge 0 and a>0 or w>0 or f>0 or o>0 then do;
        M94=M94+1;
      end;
      if a>1 then M94=-3;
      if w>1 then M94=-3;
      if f>1 then M94=-3;
      if o>1 then M94=-3;
      array allamt4 aamt wamt famt oamt;
      do over allamt4;
        if -3 le allamt4 le -1 then do;
          nn4=nn4+1;
          amt94=allamt4;
        end;
        if allamt4 ge 0 then do;
          na4=na4+1;
          if na4=1 and nn4=0 then amt94=allamt4;

```

```

if na4>1 and nn4=0 then amt94=amt94+allamt4;
  if M94=-3 then amt94=-3;
end;
if UM94 ge 0 and UI>0 then do;
  UM94=UM94+1;
end;
if UI>1 then UM94=-3;
if -3 le uiamt le -1 then do;
  un4=un4+1;
  uamt94=uamt;
end;
if uiamt ge 0 then do;
  ua4=ua4+1;
  if ua4=1 and un4=0 then UAMT94=UIAMT;
  if ua4>1 and un4=0 then uamt94=uamt94+uamt;
  if UM94=-3 then uamt94=-3;
end;
if WM94 ge 0 and WC>0 then do;
  WM94=WM94+1;
end;
if WC>1 then WM94=-3;
if -3 le wcamt le -1 then do;
  wn4=wn4+1;
  camt94=wcamt;
end;
if wcamt ge 0 then do;
  wa4=wa4+1;
  if wa4=1 and wn4=0 then cAMT94=WCAMT;
  if wa4>1 and wn4=0 then
    camt94=camt94+wcamt;
  if WM94=-3 then camt94=-3;
end;
end;

if 181 le L le 192 then do;
  if M95 ge 0 and a>0 or w>0 or f>0 or o>0 then do;
    M95=M95+1;
  end;
  if a>1 then M95=-3;
  if w>1 then M95=-3;
  if f>1 then M95=-3;
  if o>1 then M95=-3;
  array allamt5 aamt wamt famt oamt;
  do over allamt5;
    if -3 le allamt5 le -1 then do;
      nn5=nn5+1;
      amt95=allamt5;
    end;
    if allamt5 ge 0 then do;
      na5=na5+1;
      if na5=1 and nn5=0 then amt95=allamt5;
      if na5>1 and nn5=0 then amt95=amt95+allamt5;
      if M95=-3 then amt95=-3;
    end;
  end;
  if UM95 ge 0 and UI>0 then do;
    UM95=UM95+1;
  end;
  if UI>1 then UM95=-3;
  if -3 le uiamt le -1 then do;
    un5=un5+1;
    uamt95=uiamt;
  end;
  if uiamt ge 0 then do;
    ua5=ua5+1;
    if ua5=1 and un5=0 then UAMT95=UIAMT;
    if ua5>1 and un5=0 then uamt95=uamt95+uamt;
    if UM95=-3 then uamt95=-3;
  end;
  if WM95 ge 0 and WC>0 then do;
    WM95=WM95+1;
  end;
  if WC>1 then WM95=-3;
  if -3 le wcamt le -1 then do;
    wn5=wn5+1;
    camt95=wcamt;
  end;
  if wcamt ge 0 then do;
    wa5=wa5+1;
    if wa5=1 and wn5=0 then cAMT95=WCAMT;
    if wa5>1 and wn5=0 then
      camt95=camt95+wcamt;
    if WM95=-3 then camt95=-3;
  end;
end;

if 193 le L le 204 then do;
  if M96 ge 0 and a>0 or w>0 or f>0 or o>0 then do;
    M96=M96+1;
  end;
  if a>1 then M96=-3;
  if w>1 then M96=-3;
  if f>1 then M96=-3;
  if o>1 then M96=-3;
  array allamt6 aamt wamt famt oamt;
  do over allamt6;
    if -3 le allamt6 le -1 then do;
      nn6=nn6+1;
      amt96=allamt6;
    end;
    if allamt6 ge 0 then do;
      na6=na6+1;
      if na6=1 and nn6=0 then amt96=allamt6;
      if na6>1 and nn6=0 then amt96=amt96+allamt6;
      if M96=-3 then amt96=-3;
    end;
  end;
  if UM96 ge 0 and UI>0 then do;
    UM96=UM96+1;
  end;
  if UI>1 then UM96=-3;
  if -3 le uiamt le -1 then do;
    un6=un6+1;
  end;

```

<pre> uamt96=uiamt; end; if uiamt ge 0 then do; ua6=ua6+1; if ua6=1 and un6=0 then UAMT96=UIAMT; if ua6>1 and un6=0 then uamt96=uamt96+uiamt; if UM96=-3 then uamt96=-3; end; if WM96 ge 0 and WC>0 then do; WM96=WM96+1; end; if WC>1 then WM96=-3; if -3 le wcamt le -1 then do; wn6=wn6+1; camt96=wcamt; end; if wcamt ge 0 then do; wa6=wa6+1; if wa6=1 and wn6=0 then cAMT96=WCAMT; if wa6>1 and wn6=0 then camt96=camt96+wcamt; if WM96=-3 then camt96=-3; end; end; if 205 le L le 216 then do; if M97 ge 0 and a>0 or w>0 or f>0 or o>0 then do; M97=M97+1; end; if a>1 then M97=-3; if w>1 then M97=-3; if f>1 then M97=-3; if o>1 then M97=-3; array allamt7 aamt wamt famt oamt; do over allamt7; if -3 le allamt7 le -1 then do; nn7=nn7+1; amt97=allamt7; end; if allamt7 ge 0 then do; na7=na7+1; if na7=1 and nn7=0 then amt97=allamt7; if na7>1 and nn7=0 then amt97=amt97+allamt7; if M97=-3 then amt97=-3; end; end; if UM97 ge 0 and UI>0 then do; UM97=UM97+1; end; if UI>1 then UM97=-3; if -3 le uiamt le -1 then do; un7=un7+1; uamt97=uiamt; end; if uiamt ge 0 then do; ua7=ua7+1; if ua7=1 and un7=0 then UAMT97=UIAMT; </pre>	<pre> if ua7>1 and un7=0 then uamt97=uamt97+uiamt; if UM97=-3 then uamt97=-3; end; if WM97 ge 0 and WC>0 then do; WM97=WM97+1; end; if WC>1 then WM97=-3; if -3 le wcamt le -1 then do; wn7=wn7+1; camt97=wcamt; end; if wcamt ge 0 then do; wa7=wa7+1; if wa7=1 and wn7=0 then cAMT97=WCAMT; if wa7>1 and wn7=0 then camt97=camt97+wcamt; if WM97=-3 then camt97=-3; end; end; if 217 le L le 228 then do; if M98 ge 0 and a>0 or w>0 or f>0 or o>0 then do; M98=M98+1; end; if a>1 then M98=-3; if w>1 then M98=-3; if f>1 then M98=-3; if o>1 then M98=-3; array allamt8 aamt wamt famt oamt; do over allamt8; if -3 le allamt8 le -1 then do; nn8=nn8+1; amt98=allamt8; end; if allamt8 ge 0 then do; na8=na8+1; if na8=1 and nn8=0 then amt98=allamt8; if na8>1 and nn8=0 then amt98=amt98+allamt8; if M98=-3 then amt98=-3; end; end; if UM98 ge 0 and UI>0 then do; UM98=UM98+1; end; if UI>1 then UM98=-3; if -3 le uiamt le -1 then do; un8=un8+1; uamt98=uiamt; end; if uiamt ge 0 then do; ua8=ua8+1; if ua8=1 and un8=0 then UAMT98=UIAMT; if ua8>1 and un8=0 then uamt98=uamt98+uiamt; if UM98=-3 then uamt98=-3; end; if WM98 ge 0 and WC>0 then do; WM98=WM98+1; </pre>
--	--

```

end;
if WC>1 then WM98=-3;
if -3 le wcamt le -1 then do;
  wn8=wn8+1;
  camt98=wcamt;
end;
if wcamt ge 0 then do;
  wa8=wa8+1;
  if wa8=1 and wn8=0 then cAMT98=WCAMT;
  if wa8>1 and wn8=0 then
    camt98=camt98+wcamt;
  if WM98=-3 then camt98=-3;
end;
end;
if 229 le L le 232 then do;
  if M99 ge 0 and a>0 or w>0 or f>0 or o>0 then do;
    M99=M99+1;
  end;
  if a>1 then M99=-3;
  if w>1 then M99=-3;
  if f>1 then M99=-3;
  if o>1 then M99=-3;

array allamt9 aamt wamt famt oamt;
do over allamt9;
  if -3 le allamt9 le -1 then do;
    nn9=nn9+1;
    amt99=allamt9;
  end;
  if allamt9 ge 0 then do;
    na9=na9+1;
    if na9=1 and nn9=0 then amt99=allamt9;
    if na9>1 and nn9=0 then amt99=amt99+allamt9;
    if M99=-3 then amt99=-3;
  end;
end;

if UM99 ge 0 and UI>0 then do;
  UM99=UM99+1;
end;
if UI>1 then UM99=-3;
if -3 le uiamt le -1 then do;
  un9=un9+1;
  uamt99=uiamt;
end;
if uiamt ge 0 then do;
  ua9=ua9+1;
  if ua9=1 and un9=0 then UAMT99=UIAMT;
  if ua9>1 and un9=0 then uamt99=uamt99+uiamt;
  if UM99=-3 then uamt99=-3;
end;
if WM99 ge 0 and WC>0 then do;
  WM99=WM99+1;
end;
if WC>1 then WM99=-3;
if -3 le wcamt le -1 then do;
  wn9=wn9+1;

```

```

camt99=wcamt;
end;
if wcamt ge 0 then do;
  wa9=wa9+1;
  if wa9=1 and wn9=0 then cAMT99=WCAMT;
  if wa9>1 and wn9=0 then
    camt99=camt99+wcamt;
  if WM99=-3 then camt99=-3;
end;
end;

**for income program amounts;
if 205 le L le 216 then do;
  if w>0 or f>0 then do;
    out=1;
  end;
  if incprg ge 0 and (a>0 or o>0 or ui>0 or wc>0)
    then do;
    incprg=incprg+1;
  end;
  if a>1 or o>1 or ui>1 or wc>1 then incprg=-3;
  array prgamts aamt oamt uiamt wcamt;
  do over prgamts;
    if -3 le prgamts le -1 then do;
      nnp=nnp+1;
      prgamt=prgamts;
    end;
    if prgamts ge 0 then do;
      nap=nap+1;
      if nap=1 and nnp=0 then prgamt=prgamts;
      if nap>1 and nnp=0 then
        prgamt=prgamt+prgamts;
      if incprg=-3 then prgamt=-3;
    end;
    end;
  end;
end;
/*3. end*/

if total80=. then total80=0;
if total81=. then total81=0;
if total82=. then total82=0;
if total83=. then total83=0;
if total84=. then total84=0;
if total85=. then total85=0;
if total86=. then total86=0;
if total87=. then total87=0;
if total88=. then total88=0;
if total89=. then total89=0;
if total90=. then total90=0;
if total91=. then total91=0;
if total92=. then total92=0;
if total93=. then total93=0;
if total94=. then total94=0;
if total95=. then total95=0;
if total96=. then total96=0;

```

```

if total97=. then total97=0;
if total98=. then total98=0;
if totalall=. then totalall=0;

ckM80=total80;
ckM81=total81;
ckM82=total82;
ckM83=total83;
ckM84=total84;
ckM85=total85;
ckM86=total86;
ckM87=total87;
ckM88=total88;
ckM89=total89;
ckM90=total90;
ckM91=total91;
ckM92=total92;
ckM93=total93;
ckM94=total94;
ckM95=total95;
ckM96=total96;
ckM97=total97;
ckM98=total98;

if M80 ge 0 and total80 ge 0 then do;
  total80=M80+total80; end;
if M81 ge 0 and total81 ge 0 then do;
  total81=M81+total81; end;
if M82 ge 0 and total82 ge 0 then do;
  total82=M82+total82; end;
if M83 ge 0 and total83 ge 0 then do;
  total83=M83+total83; end;
if M84 ge 0 and total84 ge 0 then do;
  total84=M84+total84; end;
if M85 ge 0 and total85 ge 0 then do;
  total85=M85+total85; end;
if M86 ge 0 and total86 ge 0 then do;
  total86=M86+total86; end;
if M87 ge 0 and total87 ge 0 then do;
  total87=M87+total87; end;
if M88 ge 0 and total88 ge 0 then do;
  total88=M88+total88; end;
if M89 ge 0 and total89 ge 0 then do;
  total89=M89+total89; end;
if M90 ge 0 and total90 ge 0 then do;
  total90=M90+total90; end;
if M91 ge 0 and total91 ge 0 then do;
  total91=M91+total91; end;
if M92 ge 0 and total92 ge 0 then do;
  total92=M92+total92; end;
if M93 ge 0 and total93 ge 0 then do;
  total93=M93+total93; end;
if M94 ge 0 and total94 ge 0 then do;
  total94=M94+total94; end;
if M95 ge 0 and total95 ge 0 then do;
  total95=M95+total95; end;

```

```

if M96 ge 0 and total96 ge 0 then do;
  total96=M96+total96; end;
if M97 ge 0 and total97 ge 0 then do;
  total97=M97+total97; end;
if M98 ge 0 and total98 ge 0 then do;
  total98=M98+total98; end;
total99=m99;

if M80<0 or total80<0 then do; total80=-3; end;
if M81<0 or total81<0 then do; total81=-3; end;
if M82<0 or total82<0 then do; total82=-3; end;
if M83<0 or total83<0 then do; total83=-3; end;
if M84<0 or total84<0 then do; total84=-3; end;
if M85<0 or total85<0 then do; total85=-3; end;
if M86<0 or total86<0 then do; total86=-3; end;
if M87<0 or total87<0 then do; total87=-3; end;
if M88<0 or total88<0 then do; total88=-3; end;
if M89<0 or total89<0 then do; total89=-3; end;
if M90<0 or total90<0 then do; total90=-3; end;
if M91<0 or total91<0 then do; total91=-3; end;
if M92<0 or total92<0 then do; total92=-3; end;
if M93<0 or total93<0 then do; total93=-3; end;
if M94<0 or total94<0 then do; total94=-3; end;
if M95<0 or total95<0 then do; total95=-3; end;
if M96<0 or total96<0 then do; total96=-3; end;
if M97<0 or total97<0 then do; total97=-3; end;
if M98<0 or total98<0 then do; total98=-3; end;
if m99<0 then do; total99=-3; end;

*total months, part 2;
if ttlm ge 0 and totalall ge 0 then do;
  ttlm=ttlm+totalall; end;
if ttlm<0 or totalall<0 then do;
  ttlm=-3; end;

ckamt80=atotal80;
ckamt81=atotal81;
ckamt82=atotal82;
ckamt83=atotal83;
ckamt84=atotal84;
ckamt85=atotal85;
ckamt86=atotal86;
ckamt87=atotal87;
ckamt88=atotal88;
ckamt89=atotal89;
ckamt90=atotal90;
ckamt91=atotal91;
ckamt92=atotal92;
ckamt93=atotal93;
ckamt94=atotal94;
ckamt95=atotal95;
ckamt96=atotal96;
ckamt97=atotal97;
ckamt98=atotal98;

if atotal80=-4 then atotal80=0;
if atotal81=-4 then atotal81=0;

```

<pre> if atotal82=-4 then atotal82=0; if atotal83=-4 then atotal83=0; if atotal84=-4 then atotal84=0; if atotal85=-4 then atotal85=0; if atotal86=-4 then atotal86=0; if atotal87=-4 then atotal87=0; if atotal88=-4 then atotal88=0; if atotal89=-4 then atotal89=0; if atotal90=-4 then atotal90=0; if atotal91=-4 then atotal91=0; if atotal92=-4 then atotal92=0; if atotal93=-4 then atotal93=0; if atotal94=-4 then atotal94=0; if atotal95=-4 then atotal95=0; if atotal96=-4 then atotal96=0; if atotal97=-4 then atotal97=0; if atotal98=-4 then atotal98=0; if amt80 ge 0 and atotal80 ge 0 then do; atotal80=amt80+atotal80; end; if amt81 ge 0 and atotal81 ge 0 then do; atotal81=amt81+atotal81; end; if amt82 ge 0 and atotal82 ge 0 then do; atotal82=amt82+atotal82; end; if amt83 ge 0 and atotal83 ge 0 then do; atotal83=amt83+atotal83; end; if amt84 ge 0 and atotal84 ge 0 then do; atotal84=amt84+atotal84; end; if amt85 ge 0 and atotal85 ge 0 then do; atotal85=amt85+atotal85; end; if amt86 ge 0 and atotal86 ge 0 then do; atotal86=amt86+atotal86; end; if amt87 ge 0 and atotal87 ge 0 then do; atotal87=amt87+atotal87; end; if amt88 ge 0 and atotal88 ge 0 then do; atotal88=amt88+atotal88; end; if amt89 ge 0 and atotal89 ge 0 then do; atotal89=amt89+atotal89; end; if amt90 ge 0 and atotal90 ge 0 then do; atotal90=amt90+atotal90; end; if amt91 ge 0 and atotal91 ge 0 then do; atotal91=amt91+atotal91; end; if amt92 ge 0 and atotal92 ge 0 then do; atotal92=amt92+atotal92; end; if amt93 ge 0 and atotal93 ge 0 then do; atotal93=amt93+atotal93; end; if amt94 ge 0 and atotal94 ge 0 then do; atotal94=amt94+atotal94; end; if amt95 ge 0 and atotal95 ge 0 then do; atotal95=amt95+atotal95; end; if amt96 ge 0 and atotal96 ge 0 then do; atotal96=amt96+atotal96; end; if amt97 ge 0 and atotal97 ge 0 then do; atotal97=amt97+atotal97; end; if amt98 ge 0 and atotal98 ge 0 then do; atotal98=amt98+atotal98; end; atotal99=amt99; </pre>	<pre> if amt80<0 or atotal80<0 or total80<0 then do; atotal80=-3; end; if amt81<0 or atotal81<0 or total81<0 then do; atotal81=-3; end; if amt82<0 or atotal82<0 or total82<0 then do; atotal82=-3; end; if amt83<0 or atotal83<0 or total83<0 then do; atotal83=-3; end; if amt84<0 or atotal84<0 or total84<0 then do; atotal84=-3; end; if amt85<0 or atotal85<0 or total85<0 then do; atotal85=-3; end; if amt86<0 or atotal86<0 or total86<0 then do; atotal86=-3; end; if amt87<0 or atotal87<0 or total87<0 then do; atotal87=-3; end; if amt88<0 or atotal88<0 or total88<0 then do; atotal88=-3; end; if amt89<0 or atotal89<0 or total89<0 then do; atotal89=-3; end; if amt90<0 or atotal90<0 or total90<0 then do; atotal90=-3; end; if amt91<0 or atotal91<0 or total91<0 then do; atotal91=-3; end; if amt92<0 or atotal92<0 or total92<0 then do; atotal92=-3; end; if amt93<0 or atotal93<0 or total93<0 then do; atotal93=-3; end; if amt94<0 or atotal94<0 or total94<0 then do; atotal94=-3; end; if amt95<0 or atotal95<0 or total95<0 then do; atotal95=-3; end; if amt96<0 or atotal96<0 or total96<0 then do; atotal96=-3; end; if amt97<0 or atotal97<0 or total97<0 then do; atotal97=-3; end; if amt98<0 or atotal98<0 or total98<0 then do; atotal98=-3; end; if amt80=0 and ckamt80=-4 and total80=0 then do; atotal80=-4; end; if amt81=0 and ckamt81=-4 and total81=0 then do; atotal81=-4; end; if amt82=0 and ckamt82=-4 and total82=0 then do; atotal82=-4; end; if amt83=0 and ckamt83=-4 and total83=0 then do; atotal83=-4; end; if amt84=0 and ckamt84=-4 and total84=0 then do; atotal84=-4; end; if amt85=0 and ckamt85=-4 and total85=0 then do; atotal85=-4; end; if amt86=0 and ckamt86=-4 and total86=0 then do; atotal86=-4; end; if amt87=0 and ckamt87=-4 and total87=0 then do; atotal87=-4; end; if amt88=0 and ckamt88=-4 and total88=0 then do; </pre>
--	--

```

        atotal88=-4; end;
if amt89=0 and ckamt89=-4 and total89=0 then do;
        atotal89=-4; end;
if amt90=0 and ckamt90=-4 and total90=0 then do;
        atotal90=-4; end;
if amt91=0 and ckamt91=-4 and total91=0 then do;
        atotal91=-4; end;
if amt92=0 and ckamt92=-4 and total92=0 then do;
        atotal92=-4; end;
if amt93=0 and ckamt93=-4 and total93=0 then do;
        atotal93=-4; end;
if amt94=0 and ckamt94=-4 and total94=0 then do;
        atotal94=-4; end;
if amt95=0 and ckamt95=-4 and total95=0 then do;
        atotal95=-4; end;
if amt96=0 and ckamt96=-4 and total96=0 then do;
        atotal96=-4; end;
if amt97=0 and ckamt97=-4 and total97=0 then do;
        atotal97=-4; end;
if amt98=0 and ckamt98=-4 and total98=0 then do;
        atotal98=-4; end;
if amt99=0 and total99=0 then do;
        atotal99=-4; end;

if uamt80=0 and um80=0 then do; uamt80=-4; end;
if uamt81=0 and um81=0 then do; uamt81=-4; end;
if uamt82=0 and um82=0 then do; uamt82=-4; end;
if uamt83=0 and um83=0 then do; uamt83=-4; end;
if uamt84=0 and um84=0 then do; uamt84=-4; end;
if uamt85=0 and um85=0 then do; uamt85=-4; end;
if uamt86=0 and um86=0 then do; uamt86=-4; end;
if uamt87=0 and um87=0 then do; uamt87=-4; end;
if uamt88=0 and um88=0 then do; uamt88=-4; end;
if p1210=-4 then do;

total80=-4; total81=-4; total82=-4; total83=-4; total84=-4; total85=-4; total86=-4; total87=-4;
total88=-4; total89=-4; total90=-4; total91=-4; total92=-4; total93=-4; total94=-4; total95=-4;
total96=-4; total97=-4; total98=-4; total99=-4;
atotal80=-4; atotal81=-4; atotal82=-4; atotal83=-4; atotal84=-4; atotal85=-4; atotal86=-4; atotal87=-4;
atotal88=-4; atotal89=-4; atotal90=-4; atotal91=-4; atotal92=-4; atotal93=-4; atotal94=-4; atotal95=-4;
atotal96=-4; atotal97=-4; atotal98=-4; atotal99=-4;
m80=-4; m81=-4; M82=-4; M83=-4; M84=-4; M85=-4; M86=-4; M87=-4; M88=-4; M89=-4;
m90=-4; m91=-4; M92=-4; M93=-4; M94=-4; M95=-4; M96=-4; M97=-4; M98=-4; M99=-4; ttlM=-4;
amt80=-4; amt81=-4; amt82=-4; amt83=-4; amt84=-4; amt85=-4; amt86=-4; amt87=-4; amt88=-4; amt89=-4;
amt90=-4; amt91=-4; amt92=-4; amt93=-4; amt94=-4; amt95=-4; amt96=-4; amt97=-4; amt98=-4; amt99=-4;
UM80=-4; UM81=-4; UM82=-4; UM83=-4; UM84=-4; UM85=-4; UM86=-4; UM88=-4; UM87=-4;
UM88=-4; UM89=-4; UM90=-4; UM91=-4; UM92=-4; UM93=-4; UM94=-4; UM95=-4; UM96=-4;
UM97=-4; UM98=-4; UM99=-4;
WM80=-4; WM81=-4; WM82=-4; WM83=-4; WM84=-4; WM85=-4; WM86=-4; WM88=-4; WM87=-4;
WM88=-4; WM89=-4; WM90=-4; WM91=-4; WM92=-4; WM93=-4; WM94=-4; WM95=-4; WM96=-4;
WM97=-4; WM98=-4; WM99=-4; TTLU=-4; TTLW=-4;
camt80=-4; camt81=-4; camt82=-4; camt83=-4; camt84=-4; camt85=-4; camt86=-4; camt87=-4; camt88=-4;
camt89=-4; camt90=-4; camt91=-4; camt92=-4; camt93=-4; camt94=-4; camt95=-4; camt96=-4; camt97=-4;
camt98=-4; camt99=-4;
uamt80=-4; uamt81=-4; uamt82=-4; uamt83=-4; uamt84=-4; uamt85=-4; uamt86=-4; uamt87=-4; uamt88=-4;
uamt89=-4; uamt90=-4; uamt91=-4; uamt92=-4; uamt93=-4; uamt94=-4; uamt95=-4; uamt96=-4; uamt97=-4;
uamt98=-4; uamt99=-4;

```

```

na80=-4; nn80=-4; na81=-4; nn81=-4; na82=-4; nn82=-4; na83=-4; nn83=-4; na84=-4; nn84=-4; na85=-4;
nn85=-4; na86=-4; nn86=-4; na87=-4; nn87=-4; na88=-4; nn88=-4; na89=-4; nn89=-4;
na0=-4; nn0=-4; na1=-4; nn1=-4; na2=-4; nn2=-4; na3=-4; nn3=-4;
na4=-4; nn4=-4; na5=-4; nn5=-4; na6=-4; nn6=-4; na7=-4; nn7=-4;
na8=-4; nn8=-4; na9=-4; nn9=-4;
ua80=-4; un80=-4; ua81=-4; un81=-4;
wa80=-4; wn80=-4; wa81=-4; wn81=-4;
ua82=-4; un82=-4; ua83=-4; un83=-4; ua84=-4; un84=-4;
wa82=-4; wn82=-4; wa83=-4; wn83=-4; wa84=-4; wn84=-4;
ua85=-4; un85=-4; ua86=-4; un86=-4; ua87=-4; un87=-4;
wa85=-4; wn85=-4; wa86=-4; wn86=-4; wa87=-4; wn87=-4;
ua88=-4; un88=-4; ua89=-4; un89=-4; ua0=-4; un0=-4;
wa88=-4; wn88=-4; wa89=-4; wn89=-4; wa0=-4; wn0=-4;
ua1=-4; un1=-4; ua2=-4; un2=-4; ua3=-4; un3=-4;
wa1=-4; wn1=-4; wa2=-4; wn2=-4; wa3=-4; wn3=-4;
ua4=-4; un4=-4; ua5=-4; un5=-4; ua6=-4; un6=-4;
wa4=-4; wn4=-4; wa5=-4; wn5=-4; wa6=-4; wn6=-4;
ua7=-4; un7=-4; ua8=-4; un8=-4; ua9=-4; un9=-4;
wa7=-4; wn7=-4; wa8=-4; wn8=-4; wa9=-4; wn9=-4;
incprg=-4; prgamt=-4; nnp=-4; nap=-4; out=-4;
NU80=-4; NU81=-4; NU82=-4; NU83=-4; NU84=-4; NU85=-4; NU86=-4; NU87=-4; NU88=-4; NU89=-4;
NU90=-4; NU91=-4; NU92=-4; NU93=-4; NU94=-4; NU95=-4; NU96=-4; NU97=-4; NU98=-4; NU99=-4;

end;

if p1210=-5 then do;
total80=-5; total81=-5; total82=-5; total83=-5; total84=-5; total85=-5; total86=-5; total87=-5;
total88=-5; total89=-5; total90=-5; total91=-5; total92=-5; total93=-5; total94=-5; total95=-5;
total96=-5; total97=-5; total98=-5; total99=-5;
atotal80=-5; atotal81=-5; atotal82=-5; atotal83=-5; atotal84=-5; atotal85=-5; atotal86=-5; atotal87=-5;
atotal88=-5; atotal89=-5; atotal90=-5; atotal91=-5; atotal92=-5; atotal93=-5; atotal94=-5; atotal95=-5;
atotal96=-5; atotal97=-5; atotal98=-5; atotal99=-5;
m80=-5; m81=-5; M82=-5; M83=-5; M84=-5; M85=-5; M86=-5; M87=-5; M88=-5; M89=-5;
m90=-5; m91=-5; M92=-5; M93=-5; M94=-5; M95=-5; M96=-5; M97=-5; M98=-5; M99=-5; ttlM=-5;
amt80=-5; amt81=-5; amt82=-5; amt83=-5; amt84=-5; amt85=-5; amt86=-5; amt87=-5; amt88=-5; amt89=-5;
amt90=-5; amt91=-5; amt92=-5; amt93=-5; amt94=-5; amt95=-5; amt96=-5; amt97=-5; amt98=-5; amt99=-5;
UM80=-5; UM81=-5; UM82=-5; UM83=-5; UM84=-5; UM85=-5; UM86=-5; UM88=-5; UM87=-5;
UM88=-5; UM89=-5; UM90=-5; UM91=-5; UM92=-5; UM93=-5; UM94=-5; UM95=-5; UM96=-5;
UM97=-5; UM98=-5; UM99=-5;
WM80=-5; WM81=-5; WM82=-5; WM83=-5; WM84=-5; WM85=-5; WM86=-5; WM88=-5; WM87=-5;
WM88=-5; WM89=-5; WM90=-5; WM91=-5; WM92=-5; WM93=-5; WM94=-5; WM95=-5; WM96=-5;
WM97=-5; WM98=-5; WM99=-5; TTLU=-5; TTLW=-5;
camt80=-5; camt81=-5; camt82=-5; camt83=-5; camt84=-5; camt85=-5; camt86=-5; camt87=-5; camt88=-5;
camt89=-5; camt90=-5; camt91=-5; camt92=-5; camt93=-5; camt94=-5; camt95=-5; camt96=-5; camt97=-5;
camt98=-5; camt99=-5;
uamt80=-5; uamt81=-5; uamt82=-5; uamt83=-5; uamt84=-5; uamt85=-5; uamt86=-5; uamt87=-5; uamt88=-5;
uamt89=-5; uamt90=-5; uamt91=-5; uamt92=-5; uamt93=-5; uamt94=-5; uamt95=-5; uamt96=-5; uamt97=-5;
uamt98=-5; uamt99=-5;
na80=-5; nn80=-5; na81=-5; nn81=-5; na82=-5; nn82=-5; na83=-5; nn83=-5; na84=-5; nn84=-5; na85=-5;
nn85=-5; na86=-5; nn86=-5; na87=-5; nn87=-5; na88=-5; nn88=-5; na89=-5; nn89=-5;
na0=-5; nn0=-5; na1=-5; nn1=-5; na2=-5; nn2=-5; na3=-5; nn3=-5;
na4=-5; nn4=-5; na5=-5; nn5=-5; na6=-5; nn6=-5; na7=-5; nn7=-5;
na8=-5; nn8=-5; na9=-5; nn9=-5;
ua80=-5; un80=-5; ua81=-5; un81=-5;
wa80=-5; wn80=-5; wa81=-5; wn81=-5;
ua82=-5; un82=-5; ua83=-5; un83=-5; ua84=-5; un84=-5;
wa82=-5; wn82=-5; wa83=-5; wn83=-5; wa84=-5; wn84=-5;

```

```
ua85=-5; un85=-5; ua86=-5; un86=-5; ua87=-5; un87=-5;
wa85=-5; wn85=-5; wa86=-5; wn86=-5; wa87=-5; wn87=-5;
ua88=-5; un88=-5; ua89=-5; un89=-5; ua0=-5; un0=-5;
wa88=-5; wn88=-5; wa89=-5; wn89=-5; wa0=-5; wn0=-5;
ua1=-5; un1=-5; ua2=-5; un2=-5; ua3=-5; un3=-5;
wa1=-5; wn1=-5; wa2=-5; wn2=-5; wa3=-5; wn3=-5;
ua4=-5; un4=-5; ua5=-5; un5=-5; ua6=-5; un6=-5;
wa4=-5; wn4=-5; wa5=-5; wn5=-5; wa6=-5; wn6=-5;
ua7=-5; un7=-5; ua8=-5; un8=-5; ua9=-5; un9=-5;
wa7=-5; wn7=-5; wa8=-5; wn8=-5; wa9=-5; wn9=-5;
incprg=-5; prgamt=-5; nnp=-5; nap=-5; out=-5;
NU80=-5; NU81=-5; NU82=-5; NU83=-5; NU84=-5; NU85=-5; NU86=-5; NU87=-5; NU88=-5; NU89=-5;
NU90=-5; NU91=-5; NU92=-5; NU93=-5; NU94=-5; NU95=-5; NU96=-5; NU97=-5; NU98=-5; NU99=-5;

end;
endsas;
```